

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

APPROCHE ONTOLOGIQUE POUR LA GESTION DES CONNAISSANCES
DANS UNE ORGANISATION

MÉMOIRE
PRÉSENTÉ
COMME EXIGENCE PARTIELLE
DE LA MAÎTRISE EN INFORMATIQUE DE GESTION

PAR
KARIM SEDDIKI

AOÛT 2007

UNIVERSITÉ DU QUÉBEC À MONTRÉAL
Service des bibliothèques

Avertissement

La diffusion de ce mémoire se fait dans le respect des droits de son auteur, qui a signé le formulaire *Autorisation de reproduire et de diffuser un travail de recherche de cycles supérieurs* (SDU-522 – Rév.01-2006). Cette autorisation stipule que «conformément à l'article 11 du Règlement no 8 des études de cycles supérieurs, [l'auteur] concède à l'Université du Québec à Montréal une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de [son] travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, [l'auteur] autorise l'Université du Québec à Montréal à reproduire, diffuser, prêter, distribuer ou vendre des copies de [son] travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris l'Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de [la] part [de l'auteur] à [ses] droits moraux ni à [ses] droits de propriété intellectuelle. Sauf entente contraire, [l'auteur] conserve la liberté de diffuser et de commercialiser ou non ce travail dont [il] possède un exemplaire.»

À mes parents :
Ma mère Hamida Djaïd
et
Mon père Yahya Seddiki

REMERCIEMENT

Je tiens à remercier M. Tho Hau Nguyen professeur à l'université du Québec à Montréal, qui a accepté de diriger ce travail de recherche, pour ces remarques constructives et la confiance qu'il a témoignée tout au long de cette recherche

Je remercie aussi M. Yves Lambert médecin et professeur adjoint à l'université de Montréal, qui a accepté de codiriger ce travail et qui m'a introduit au centre de jeunesse de la Montérégie.

Je remercie également les membres du jury Monsieur Gilles Gauthier et Monsieur Lorne H. Bouchard qui ont eu l'amabilité d'évaluer ce mémoire. Cette évaluation qui a duré exceptionnellement plus de six mois.

Merci à mon amie Pascale Noutsu qui m'a aidé à améliorer la qualité du français de ce document.

Merci à mes parents et tous mes frères et sœurs pour leur soutien et leur encouragement.

Merci à mon ami Noureddine Zouaoui pour ces encouragements et son soutien moral.

Merci à tous mes collègues de laboratoire d'informatique de gestion pour leur soutien et leurs remarques constructives.

TABLE DES MATIÈRES

LISTE DES FIGURES.....	viii
LISTE DES TABLEAUX.....	x
LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES.....	xi
RÉSUMÉ.....	xiii
INTRODUCTION.....	1
CHAPITRE I	
PROBÉMATIQUE DE LA RECHERCHE.....	6
1.1 Le centre jeunesse de la Montérégie (CJM).....	6
1.2 La mission du centre jeunesse de la Montérégie.....	7
1.2.1 La protection.....	7
1.2.2 Le traitement.....	7
1.2.3 La prévention et la promotion.....	7
1.2.4 La compréhension des problèmes.....	7
1.3 Les services offerts au CJM.....	7
1.3.1 Service de protection de la jeunesse.....	7
1.3.2 Services psychosociaux.....	9
1.3.3 Services d'hébergement.....	9
1.3.4 Service de réadaptation.....	9
1.3.5 Service particulier / Adoption québécoise et internationale.....	9
1.4 Description du domaine des thérapies familiales.....	10
1.4.1 Définition de la famille.....	10
1.4.2 Les thérapies comportementalistes.....	11
1.4.3 Les thérapies psychanalytiques.....	11
1.4.4 Les thérapies familiales systémiques.....	12

1.4.5 Description du génogramme.....	13
1.5 Problématiques.....	16
1.6 L'objectif du mémoire.....	18

CHAPITRE II

MÉTHODOLOGIE DE RECHERCHE.....19

2.1 La première phase : définition du projet.....	19
2.1.1 Identification du thème de la recherche.....	19
2.1.2 Revue de la littérature et entretiens exploratoires.....	20
2.1.3 Définition du projet.....	20
2.2 La deuxième phase : Revue de la littérature et collecte de données	20
2.2.1 La recherche bibliographique.....	20
2.2.2 Les entretiens avec le médecin en titre au CJM.....	21
2.3 La troisième phase : Analyse de données et construction de l'ontologie OCJM.....	21
2.4 La quatrième phase: Développement du prototype du SGC.....	22
2.4.1 Phase d'analyse et de conception.....	22
2.4.2 Phase de réalisation.....	22
2.4.3 Phase de livraison.....	23

CHAPITRE III

LA GESTION DES CONNAISSANCES.....25

3.1 La connaissance dans l'entreprise.....	26
3.2 La gestion des connaissances.....	27
3.3 Mémoire de l'entreprise.....	30
3.4 Systèmes de gestion des connaissances.....	32
3.5 Les ontologies et la gestion de connaissances.....	32

CHAPITRE IV

LES ONTOLOGIES.....34

4.1 Définitions.....	34
4.1.1 Définitions descriptives.....	35
4.1.2 Définitions basées sur la logique.....	35

4.1.3 Définitions perspectives.....	36
4.2 Les composantes d'une ontologie.....	36
4.2.1 Modélisation des ontologies par les cadres et la logique du premier ordre.....	36
4.2.2 Modélisation des ontologies par la logique de description.....	38
4.3 Typologie d'ontologies.....	39
4.3.1 Classification selon l'objet de conceptualisation.....	40
4.3.2 Classification selon le degré de formalisation.....	40
4.4 Rôles des ontologies.....	41
4.5 Approches de développement des ontologies.....	42
4.5.1 La méthodologie de Uschold et al.....	46
4.5.2 La méthodologie « Methontology ».....	47
4.5.3 La méthodologie de Natalya et McGuinness	48

CHAPITRE V

LANGAGE D'ONTOLOGIES.....	49
5.1 Les paradigmes de représentation des connaissances.....	49
5.1.1 La logique de premier ordre.....	49
5.1.2 Le paradigme de cadres « frames ».....	50
5.1.3 La logique de description « description logic » (DL).....	50
5.1.4 Le paradigme des graphes conceptuels.....	51
5.2 Les langages traditionnels.....	51
5.2.1 Knowledge interchange format (KIF).....	51
5.2.2 Ontolingua.....	51
5.2.3 Loom	52
5.2.4 OCML (Operational Conceptual Modeling Language).....	52
5.2.5 FLOGIC (Frame Logic).....	52
5.3 Les langages Web.....	52
5.3.1 Resource Description Framework (RDF).....	52
5.3.2 DAML+OIL... ..	56
5.3.3 Le Web Ontology Language (OWL).....	57
5.3.4 Différences entre DAML+OIL et OWL.....	65

CHAPITRE VI

L'ONTOLOGIE OCJM.....	67
6.1 Conceptualisation de l'ontologie OCJM.....	67
6.2 La formalisation de l'ontologie.....	70
6.2.1 L'en-tête et les espaces de nom.....	72
6.2.2 Définition de classes.....	73
6.2.3 Définition de propriétés.....	74
6.2.4 Définition des instances.....	74

CHAPITRE VII

IMPLÉMENTATION DU PROTOTYPE.....	75
7.1 Modules du prototype.....	75
7.2 L'architecture technique du SGC.....	76
7.3 Les technologies et langages utilisés.....	76
7.4 Le déploiement du SGC.....	81
7.5 Guide d'utilisation du prototype.....	82
CONCLUSION.....	96
BIBLIOGRAPHIE	128
APPENDICE A : Exemple de principaux termes utilisés au CJM.....	99
APPENDICE B : Exemple : La classes « Génogramme ».....	100
APPENDICE C : Exemple : La classes « Personne »	102
APPENDICE D : Fichier « OCJM.owl ».....	103
APPENDICE E : Les principales classes java.....	113
APPENDICE F : Les principales pages JSP.....	123

LISTE DE FIGURES

FIGURE 1.1 : Exemple de symboles du génogramme.....	15
FIGURE 1.1 : Approche méthodologique.....	24
FIGURE 3.1 : Les modes de conversion des connaissances.....	26
FIGURE 3.2 : Cadre conceptuel de la GC.....	30
FIGURE 4.1 : Thésaurus, Taxonomie et Ontologie.....	39
FIGURE 4.2 : Les rôles des ontologies.....	42
FIGURE 5.1 : Exemple d'un graphe étiqueté orienté.....	54
FIGURE 5.2 : Déclaration d'un triplet RDF.....	55
FIGURE 5.3 : Déclaration d'espace de nom en OWL.....	59
FIGURE 5.4 : Exemple d'un document doctype.....	60
FIGURE 5.5 : Exemple d'en-tête d'ontologie OWL.....	61
FIGURE 6.1 : Les sections de l'ontologie OCJM.....	67
FIGURE 6.2 : Les sections de l'ontologie OCJM retenues.....	68
FIGURE 6.3 : Exemple d'hierarchie de classes.....	71
FIGURE 6.4 : L'espace de nom de l'ontologie OCJM.....	72
FIGURE 6.5 : L'en-tête de l'ontologie OCJM.....	72
FIGURE 6.6 : Exemple de déclaration de classes.....	73
FIGURE 7.1 : Les composantes du SGC.....	77
FIGURE 7.2 : L'architecture technique du SGC.....	78
FIGURE 7.3 : Schéma général du site.....	83
FIGURE 7.4 : Navigation dans le site.....	84
FIGURE 7.5 : La page d'accueil.....	85
FIGURE 7.6 : La page « login ».....	86
FIGURE 7.7 : Un utilisateur inscrit.....	87
FIGURE 7.8 : Début de la navigation.....	88
FIGURE 7.9 : Exemple de sous-classes.....	89
FIGURE 7.10: Exemple de consultation de l'annuaire	90

FIGURE 7.11: Choix d'une sous-classe.....	91
FIGURE 7.12: Retour à la classe mère.....	92
FIGURE 7.13: Utilisateur avec privilège	93
FIGURE 7.14: Enregistrement d'un nouvel utilisateur.....	94
FIGURE 7.15 : Chargement de l'Ontologie avec succès.....	95

LISTE DES TABLEAUX

Table 1.1 : Comparaison des approches thérapeutiques	13
Table 1.2 : Classification des liens affectifs.....	17
Table 3.1 : Les quatre catégories des définitions de la GC.....	28
Table 3.2 : Les quatre types de mémoire d'entreprise.....	31
Table 4.1 : Les trois catégories de définition d'ontologies.....	37
Table 4.2 : Les rôles des ontologies.....	43
Table 4.3 : Comparaison des principales méthodologies d'ontologies.....	45
Table 5.1 : Primitives DAML+OIL renommées en OWL.....	66
Table 6.1 : Exemple de termes utilisés au CJM.....	70
Table A.1 : Termes utilisés au CJM.....	99
Table B.1 : La classe « Génogramme ».....	100
Table C.1 : La classe « Personne ».....	102

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

API	Application Programming Interface
BD	Base de données
CJM	Centre de jeunesse de la Montérégie
CJs	Centres de jeunesse
CSSS	Centre de santé et de services sociaux
DAML	DAPRA Agent Markup Language
DL	Description Logic
GC	Gestion des connaissances
GPL	General Public Licence
HTML	Hypertext Markup Language
IDE	Integrated Development Environment
JDBC	Java Data Base Connectivity
JRE	Java Runtime Environment
JVM	Java Virtual Machine
JSP	Java Server Pages

KIF	Knowledge Interchange Format
LPJ	Loi sur la protection de la jeunesse
OIL	Ontology Inference Layer
OCJM	Ontologie du centre de jeunesse de la Montérégie
OCML	Operational Conceptual Modeling Language
OQLF	Office québécois de la langue française
OWL	Ontology Web Language
PDG	Président directeur général
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SGC	Système de gestion des connaissances
SI	Système d'information
TI	Technologie de l'information
URI	Uniform Resource Indicator
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

RÉSUMÉ

De nos jours, les organisations sont de plus en plus conscientes de l'importance de la gestion des connaissances (GC) qui consiste à créer, à acquérir, à transférer et à utiliser les connaissances dans le but d'améliorer le rendement de l'organisation. Cependant, du fait de leur formation et de leurs expériences différentes, les employés d'une même entreprise utilisent un vocabulaire et des jargons différents. Ils peuvent aussi émettre des points de vue divergents sur un même sujet. De plus, dans un tel milieu multidisciplinaire, l'intégration des nouveaux employés pose des défis majeurs. En effet, un nouvel employé qui arrive doit communiquer avec ses collègues verbalement ou par écrit, donc il devrait connaître tous les acronymes, les termes synonymes et les termes polysémiques utilisés dans son entourage. Le nouvel employé est aussi amené à lire des rapports, des manuels et d'autres documents sur différents sujets qui nécessitent certaines connaissances de base pour être compris. Dans ce contexte, les ontologies en tant qu'«une représentation explicite d'une compréhension partagée» ont immergé afin d'améliorer la communication entre les personnes et d'aider à la résolution des problèmes de synonymie et de polysémie.

Par ailleurs, l'Internet en particulier le World Wide Web est la principale infrastructure pour l'échange et la recherche d'informations. Cependant, vu l'hétérogénéité des systèmes d'informations, les informations sont souvent accompagnées de bruit (informations non pertinentes) ou de silence (manque d'information pertinente). La solution récente pour remédier à ces problèmes est le recours au « Web sémantique » qui permet de publier, de consulter et d'automatiser le traitement de connaissances formalisées. Ceci est efficacement réalisable en utilisant les ontologies. En effet, les ontologies permettent de définir les concepts à utiliser pour l'annotation sémantique des ressources afin de les capturer et de les partager entre différents systèmes et différents utilisateurs.

Ce travail a consisté à appliquer une démarche ontologique pour la gestion des connaissances dans une organisation. Cette démarche n'est pas précisément décrite dans la littérature, mais elle a été inspirée des projets de développement de systèmes de gestion de connaissances basés sur des ontologies tels que le projet *Entreprise Ontology* (Uschold et al., 1998) et le projet Gestion et Diffusion du Savoir en Télécommunication. (Lefebvre et al., 2003)

Cette démarche est supportée par un système de gestion de connaissances (SGC) à base d'ontologie. Ce SGC apporte une contribution dans les quatre domaines suivants : 1) l'amélioration de la communication entre les employés, 2) l'apprentissage des employés, 3) la recherche d'information et 4) l'intégration des nouveaux employés.

L'ontologie a été formalisée en utilisant le langage OWL, le dernier standard de l'organisation W3C. Le prototype du SGC a été réalisé en Java et JSP.

Mots clés : gestion des connaissances, ontologie, OWL

INTRODUCTION

En 1967, Drucker prédit que « Rendre les travailleurs du savoir productifs sera le principal défi des cent prochaines années, comme rendre les travailleurs manuels productifs fut celui des cent dernières. » (Davenport, 2003). Cette prédiction s'est confirmée au cours des dernières années au point que certains chercheurs ont constaté l'arrivée d'une nouvelle économie dominée par la production, l'échange et le partage de la connaissance (Zacklad et Grundstein, 2001). Dans cette économie, la gestion des connaissances (GC) ou le « knowledge management » constitue une clé importante qui permet aux organisations d'améliorer leur performance et d'obtenir un avantage concurrentiel (Ermine, 2003).

Cependant, la définition de ce qu'est la gestion des connaissances est une question délicate car les points de vue sont très divergents selon la discipline des auteurs comme nous allons le voir au chapitre 2.

Pour des raisons de clarté, nous retiendrons dans cette introduction la définition de Jacob et Pariat (2000). Selon ces deux auteurs, la gestion des connaissances est le processus qui consiste à créer, à acquérir, à transférer et à utiliser les connaissances dans le but d'améliorer le rendement de l'organisation. Cependant, de nombreuses questions se posent dans toutes les phases de ce processus. Ces questions concernent premièrement, l'identification des connaissances cruciales que l'organisation devrait gérer; et deuxièmement, la manière de conserver et d'exploiter lesdites connaissances; et finalement la façon d'échanger et partager efficacement ces connaissances entre les employés concernés.

De plus, l'échange et le partage des connaissances au sein d'une organisation passent nécessairement par une communication entre personnes et systèmes d'information (SI).

Cependant, du fait de leur formation et de leurs expériences différentes, les employés d'une même entreprise utilisent un vocabulaire et des jargons différents. Ils peuvent aussi émettre des points de vue divergents sur un même sujet. De plus, dans un tel milieu multidisciplinaire, l'intégration des nouveaux employés pose des problématiques majeures. En effet, un nouvel employé qui arrive doit communiquer avec ses collègues verbalement ou par écrit, donc il devrait connaître tous les acronymes, les termes synonymes (dont le sens est identique) et les termes polysémiques (termes ayant deux sens différents), utilisés dans son entourage. Le nouvel employé est aussi amené à lire des rapports, des manuels et d'autres documents sur différents sujets qui nécessitent certaines connaissances de base pour être compris. Cependant, l'employé ne doit pas être nécessairement spécialiste dans tous ces sujets. Dans ce contexte, les ontologies en tant qu'« une représentation explicite d'une compréhension partagée » (Gruber, 1993) ont immergé afin d'améliorer la communication entre les personnes et d'aider à la résolution des problèmes de synonymie et de polysémie (Edgington et al., 2004).

Par ailleurs, avec l'avancée de la technologie, presque toutes les entreprises utilisent l'Internet pour transférer et partager les informations et les connaissances. Donc, les employés en tant qu'utilisateurs du Web ont accès à des informations de plus en plus nombreuses. Cependant, ces informations sont souvent accompagnées de bruit (informations non pertinentes) ou de silence (manque d'information pertinente). La tendance actuelle pour remédier à ces problèmes est le recours au « Web sémantique ».

Le Web sémantique n'est pas un Web à part, mais c'est une extension du « World Wide Web » classique (Berners-Lee et al., 2001). Il permet de publier, de consulter et d'automatiser le traitement de connaissances formalisées. Pour que le Web sémantique fonctionne, les ordinateurs doivent avoir accès à des informations structurées et à un certain nombre de règles d'inférence qui seront utilisées pour parvenir à un raisonnement automatisé. Ceci est réalisable en utilisant les ontologies. En effet, les ontologies permettent de définir les

concepts à utiliser pour l'annotation sémantique des ressources afin de les capturer et de les partager entre différents systèmes et différents utilisateurs.

De nos jours, les organisations sont de plus en plus conscientes de l'importance de la gestion des connaissances et de l'apport du Web sémantique. C'est le cas du centre de jeunesse de la Montérégie (CJM), l'organisation pour laquelle nous avons effectué ce projet de recherche.

Notre projet tente de résoudre certaines problématiques en appliquant une approche de gestion des connaissances basée sur une ontologie et dans l'environnement du Web sémantique. Avant d'exposer en détails notre projet de recherche, nous présentons le CJM.

Le CJM est un exemple concret où la multidisciplinarité des employés est omniprésente. En effet, le CJM fait partie des centres de jeunesse (CJs) qui sont des établissements publics établis sur le territoire québécois. Leur mission est d'offrir une aide spécialisée aux jeunes qui connaissent des difficultés graves et à leur famille. Pour accomplir cette mission, les CJs recrutent des experts de différents domaines. Notamment des travailleurs sociaux, des psychologues, des médecins de famille, des infirmières et des préposés de bureaux.... À cause de leurs formations et de leurs expériences différentes, ces intervenants utilisent souvent des terminologies et des jargons différents. Ceci génère des confusions et de mauvaises communications en particulier chez les nouveaux employés. Cette multidisciplinarité entraîne parfois chez les intervenants une lecture différente du dossier d'un même patient. De plus, après plusieurs rencontres avec mon codirecteur, spécialiste en médecine familiale, nous avons constaté un certain nombre de problèmes notamment :

- les nouveaux employés rencontrent d'énormes difficultés à maîtriser le jargon utilisé. Ceci est dû à la multidisciplinarité des différents intervenants;
- les employés et en particulier ceux qui sont nouveaux ont de la difficulté à rechercher efficacement les informations dont ils ont besoin;
- l'absence d'un document (papier ou électronique) qui précise les règles d'utilisation des principaux termes du domaine;
- certaines relations (liens) familiales ne sont pas définies de façon précise et unanime à cause de la complexité de la structure familiale québécoise actuelle;

- l'existence de synonymies et de polysémies.

Afin de résoudre ce genre de problèmes, nous avons appliqué une démarche simple et efficace de gestion des connaissances. Cette démarche n'est pas précisément décrite dans la littérature, mais elle a été inspirée des projets de développement de systèmes de gestion des connaissances basés sur des ontologies tels que le projet *Entreprise Ontology* (Uschold et al., 1998) et le projet *Gestion et Diffusion du Savoir en Télécommunication* (Lefebvre et al., 2003)

L'objectif d'une telle démarche est d'une part de répondre aux questions concernant le processus de GC soulevées au début de cette introduction. D'autre part, il consiste à concevoir et à réaliser un système de gestion des connaissances (SGC) à base d'ontologie. Ce SGC apporte une contribution dans les quatre domaines suivants : 1) l'amélioration de la communication entre les employés, 2) l'apprentissage des employés, 3) la recherche d'informations et 4) l'intégration des nouveaux employés.

L'amélioration de la communication entre les employés est réalisée grâce à l'ontologie du centre de jeunesse de la Montérégie « OCJM » que nous avons réalisée. Cette ontologie décrit les termes utilisés dans l'organisation et les relations sémantiques entre ces termes.

L'apprentissage est supporté par le SGC qui permet à l'employé de consulter les différentes connaissances et informations cruciales structurées en hiérarchie. L'employé pourra également consulter un annuaire électronique que nous avons développé. Cet annuaire est constitué d'un certain nombre de pages Web annotées, présentant des définitions et des précisions pertinentes ainsi que des liens vers d'autres sources d'informations utiles.

Le système aide aussi les employés dans leurs recherches d'informations. En effet, l'employé trouvera l'ensemble des informations pertinentes dans la base de connaissances (l'ontologie et les instances) ce qui l'empêchera d'être noyé dans des informations dont il n'a pas besoin en consultant des sites Web ou des documents dispersés.

Finalement, la facilitation et l'accélération de l'intégration des nouveaux employés sont des conséquences directes des trois apports précédents. Cependant, ce SGC ne se substitue pas

aux efforts de l'organisation pour intégrer ses nouveaux employés (comme la formation personnalisée et les séminaires), mais il constitue un moyen efficace pour faciliter et accélérer cette intégration.

En plus de cette introduction, le reste de ce mémoire est organisé en sept chapitres

Le premier chapitre présente l'organisation pour laquelle nous avons effectué cette recherche. Il s'agit du centre de jeunesse de la Montérégie (CJM). Ce chapitre expose également la problématique, les questions de recherche traitées et les objectifs de ce mémoire.

Le deuxième chapitre, présente la méthodologie que nous avons suivie pour traiter la problématique et atteindre nos objectifs.

Le troisième chapitre présente l'état de l'art sur la gestion des connaissances des entreprises en présentant les différentes définitions, les objectifs et le rôle des ontologies dans ce domaine. Il permet ainsi de comprendre l'utilité finale des ontologies.

Le quatrième chapitre expose en détail les ontologies en décrivant notamment leurs définitions, leurs rôles et leurs approches de développement. Il est constitué de plusieurs études comparatives qui permettent d'avoir une vue complète et détaillée sur les ontologies.

Le cinquième chapitre présente les principaux langages de développement d'ontologies. Il rappelle les paradigmes de représentation des connaissances sur lesquels se basent ces langages. Il met en particulier l'emphasis sur les derniers standards de l'organisation W3C.

Le sixième chapitre présente en détail et à l'aide d'exemples, la conception et la formalisation de l'ontologie du centre de jeunesse de la Montérégie (OCJM).

Le dernier chapitre présente la conception et la réalisation de notre prototype de système de gestion des connaissances.

Enfin, la conclusion générale du mémoire sera présentée.

CHAPITRE I

PROBLÉMATIQUE DE LA RECHERCHE

Ce chapitre présente en premier lieu le centre de jeunesse de la Montérégie (CJM), l'organisation pour laquelle nous avons effectué ce projet de recherche. Ensuite, il expose la problématique et les questions de recherche que nous avons traitées ainsi que les objectifs de ce mémoire.

1.1 Le centre de jeunesse de la Montérégie (CJM) *

Le CJM est un établissement public qui offre des services spécialisés qui font appel à des expertises de pointe et qui s'adressent de façon générale aux enfants, aux jeunes en difficulté et à leur famille. Les individus concernés peuvent être âgés de 0 à 18 ans, résidants sur le territoire de la Montérégie et ayant besoin de services spécialisés. Ces derniers sont principalement de nature psychosociale et de réadaptation, comprenant ou non l'hébergement afin de répondre à des besoins intenses, persistants et de nature complexe.

Le CJM fait partie des 17 centres de jeunesse du Québec qui sont régis par la loi sur les services de santé et les services sociaux, la loi sur la protection de la jeunesse, la loi sur le système de justice pénale pour les adolescents et le code civil.

Le CJM comporte une centaine de professionnels de disciplines différentes dont la majorité sont des travailleurs sociaux, des médecins de famille, des psychoéducateurs, des psychologues et des infirmiers.

* <http://www.centrejeunessemonteregic.qc.ca>

1.2 La mission du CJM*

La mission spécifique du CJM comporte les quatre aspects suivants :

1.2.1 La protection

Le CJM intervient quand la sécurité du jeune ou celle de la société est mise en cause et quand les conditions minimales au développement de l'enfant ne sont pas rencontrées.

1.2.2 Le traitement

Le CJM traite des problèmes d'adaptation psychosociale sévères, complexes et persistants, qui nécessitent une intervention clinique spécialisée et soutenue.

1.2.3 La prévention et la promotion

Afin de prévenir l'apparition et l'aggravation de problèmes d'adaptation sociale dans certains groupes à risque, les experts du CJM interviennent tôt et avec pertinence en collaboration avec des partenaires.

1.2.4 La compréhension des problèmes

Afin d'échanger les connaissances sur les problématiques et sur les moyens d'intervention, le CJM entretient d'étroites et constantes relations avec de nombreux partenaires tels que les milieux de recherche et d'enseignement.

1.3 Les services offerts au CJM

Dans l'optique d'accomplir sa mission, le CJM offre cinq principaux services aux enfants en difficulté et à leur famille.

1.3.1 Service de protection de la jeunesse

Ce service est constitué des trois branches suivantes :

1.3.1.1 La réception des signalements et des urgences sociales

Ce service reçoit et traite tout signalement en vertu de la loi sur la protection de la jeunesse (LPJ) et détermine si la sécurité ou le développement de l'enfant est ou pourrait être considéré comme compromis.

Si le signalement n'est pas retenu, le CJM peut orienter l'enfant ou ses parents vers des services spécifiques offerts par les centres de santé et de services sociaux (CSSS).

1.3.1.2 L'évaluation/l'orientation

Ce service consiste à :

- évaluer la situation de l'enfant pour lequel un signalement a été retenu, afin de décider si sa sécurité ou son développement sont compromis;
- décider de l'orientation de l'enfant par le choix du régime volontaire ou judiciaire ainsi que par le choix des mesures appropriées;
- orienter l'enfant ou les parents vers des services spécifiques, offerts par les CSSS s'il décide que la sécurité et le développement de l'enfant ne sont pas compromis.

1.3.1.3 La révision

Ce service examine périodiquement l'ensemble de la situation d'un enfant en regard des mesures appliquées en vertu de la LPJ et détermine si des modifications sont nécessaires. C'est au moment de la révision que se prend la décision de modifications si la sécurité et le développement d'un enfant sont toujours compromis.

Les réviseurs ont donc le mandat de :

- décider de fermer ou non un dossier;
- décider si les mesures appliquées sont toujours adéquates ou doivent être modifiées;
- décider sous quel régime (volontaire ou judiciaire) il faudra procéder;
- vérifier que toutes les mesures sont prises pour assurer un retour de l'enfant chez ses parents si tel est son intérêt;
- s'assurer que l'enfant bénéficie de conditions de vie appropriées à son âge et à ses besoins;
- favoriser pour l'enfant un projet de vie stable.

* <http://www.centrejeunessemonteregie.qc.ca>

1.3.2 Services psychosociaux

Au CJM, l'intervention psychosociale est de base. Il s'agit d'une intervention individuelle, familiale ou de groupe reposant sur une perspective systémique du développement et du fonctionnement des jeunes en difficulté et de leur famille. Le CJM intervient notamment auprès du jeune en difficulté, de ses parents de son milieu social (amis, scolaire, travail, loisirs).

1.3.3 Services d'hébergement

Aux jeunes qui ne requièrent pas un milieu de vie de réadaptation spécifique, le CJM offre trois ressources d'hébergement qui sont : les familles d'accueil régulières, les familles d'accueil spécifiques et les ressources de type communautaire.

1.3.4 Services de réadaptation

C'est une intervention spécialisée, située dans le temps et dans le cadre d'objectifs formulés avec le jeune et ses parents et/ou le milieu substitut. Elle est basée sur l'approche psychoéducative et l'approche psychosociale pour intensifier et diversifier l'offre de services aux clients.

1.3.5 Services particuliers / Adoption québécoise et internationale

Le CJM offre aussi des services en matière d'adoption et de recherche des antécédents biologiques et ce, conformément à la loi sur les services de santé et les services sociaux.

Comme mentionné au début de ce chapitre, le CJM offre des services diversifiés qui nécessitent des employés spécialisés dans différents domaines. Par conséquent, les connaissances utilisées au CJM proviennent de ces différents domaines. Cependant, nous allons seulement exposer la médecine familiale pour les raisons suivantes :

- la médecine familiale est l'une des principales disciplines pratiquées au CJM;
- elle comporte des notions relativement complexes que les employés devraient connaître;
- l'étude en profondeur d'un seul domaine (la médecine familiale) nous permet de mieux comprendre les problématiques plutôt que de toucher à tous les domaines en restant dans les généralités;

- le cadre de ce travail ne permet pas d'étudier toutes les disciplines pratiquées au CJM (psychologie, sciences infirmières etc.);
- l'accessibilité des employés en médecine familiale, en particulier celle de mon codirecteur de recherche, nous était plus facile.

Pour ces raisons, nous avons donc consacré une partie de l'ontologie OCJM à la médecine familiale qui sera présentée à la section suivante.

1.4 Description du domaine des thérapies familiales

Jusqu'aux années cinquante, deux approches principales ont dominé le domaine de traitement des déviances. La première est l'approche psychanalytique fondée par S. Freud et qui met l'accent sur le passé lointain du patient. La seconde est l'approche comportementaliste qui affirme : « Faites disparaître le symptôme et vous ferez disparaître la maladie. » Au milieu des années cinquante, une nouvelle approche appelée la thérapie familiale systémique a vu le jour. Cette dernière considère la famille comme un système ouvert. Par conséquent, la déviance d'un membre est nécessairement liée à la structure et au climat familial (Maisondieu et Métayer, 1986).

Les paragraphes suivants décrivent ces trois approches thérapeutiques en se focalisant sur la thérapie familiale systémique adoptée par les spécialistes nord américains (Normand, 1993 ; Lambert, 2006). Mais tout d'abord, nous commençons par définir le concept « famille ».

1.4.1 Définitions de la famille

Au fil du temps, la famille a connu plusieurs définitions selon les points de vue adoptés. Pour le sociologue G. P. Murdock (1985) une famille désigne un « Groupe social caractérisé par la cohabitation et la coopération d'adultes des deux sexes, dont deux au moins entretiennent des relations sexuelles socialement approuvées et qui élèvent ensemble les enfants nés de cette union ou adoptés ». Il est clair que cette définition classique ne fait plus aujourd'hui l'unanimité. Elle exclut, entre autres, les cohabitations sans mariage et les familles monoparentales.

D'après l'office québécois de la langue française, une famille est un groupe d'au moins deux personnes, constitué soit d'un couple avec, le cas échéant, ses enfants célibataires de moins

de 25 ans, soit d'une personne célibataire, veuve, divorcée ou séparée et de ses enfants célibataires de moins de 25 ans (OQLF, 2005).

Dans le domaine de la thérapie familiale, la famille désigne « tout groupe qui se considère comme une famille à la condition que ce groupe comprenne des représentants d'au moins **trois générations** unies par **filiation** (Maisondieu, et Métayer, 1986). Cette définition permet de cerner la thérapie familiale des thérapies de couple et des thérapies de groupe. En effet, les thérapies de couple s'intéressent uniquement à une seule génération, celle du couple concerné; tandis que les thérapies de groupe s'appliquent à tout ensemble d'individus même si aucun lien d'affiliation n'existe entre eux.

1.4.2 Les thérapies comportementalistes

D'après les comportementalistes, le symptôme indique la persistance d'un comportement acquis dans certaines circonstances. Le malade produit encore le même comportement malgré que le contexte ait changé. Ainsi, l'objectif du thérapeute est de faire disparaître les comportements inappropriés en utilisant des techniques d'apprentissage. Cette approche, inspirée des travaux de Pavlov, affirme « faites disparaître le symptôme et vous ferez disparaître la maladie ». Dans cette approche les membres de la famille ne sont pas impliqués directement dans le processus thérapeutique, mais ils peuvent intervenir dans la phase d'apprentissage pour aider le patient à changer quelques réflexes indésirables (Maisondieu, et Métayer, 1986).

1.4.3 Les thérapies psychanalytiques

Selon cette approche, qui remonte à S. Freud, ce ne sont pas les conflits actuels qui comptent mais ce qu'ils représentent comme reviviscence de conflits passés. Par conséquent le psychanalyste cherche à découvrir, dans l'inconscient du malade, la signification des symptômes, et il n'est donc pas nécessaire, ni même souhaitable, de demander à la famille une participation active au traitement (Maisondieu et Métayer, 1986).

1.4.4 La thérapie familiale systémique

La thérapie familiale systémique désigne « toute pratique thérapeutique qui envisage la **famille** comme un ensemble **souffrant** où les symptômes de l'un de ses membres peuvent être considérés, en tout ou partie, comme en rapport avec la souffrance de tous. Quelle que soit sa référence théorique, cette technique a pour objectif de diminuer la souffrance commune en utilisant ensemble les membres de la famille comme **agents et objets** thérapeutiques» (Maisondieu et Métayer, 1986).

Par ailleurs, la thérapie familiale systémique suit une approche fondée sur « la théorie générale des systèmes » développée par le biologiste américain Ludwing von Bertalanffy dans les années 60 (Bertalanffy, 1993). Cette théorie remet en cause les bases de la rationalité cartésienne dans la recherche scientifique et dans la compréhension de la complexité. En particulier, elle remet en cause l'idée que pour répondre à une question complexe nous devons la réduire à une somme de questions simples. Bertalanffy affirme que le tout est plus que la somme des parties (Bertalanffy, 1993).

L'application de la théorie générale des systèmes aux thérapies familiales a distingué ces dernières des thérapies psychanalytiques et des thérapies comportementalistes (tabl. 1.1).

Ainsi, la thérapie familiale systémique considère la famille comme un système dont les membres ne cessent d'interagir. La famille dans cette optique s'étale sur au moins trois générations. Le thérapeute met l'accent sur la structure familiale afin de comprendre les souffrances des patients et découvrir les patterns qui se répètent de génération en génération (diabète, alcoolisme, etc.). La thérapie familiale systémique mise sur la présence, la participation et la responsabilité de la famille dans le traitement d'un de ses membres.

Tableau 1.1 : Comparaison des trois approches thérapeutiques

	Objet thérapeutique	Agent thérapeutique
Thérapies familiales systémiques	Le patient et les membres de sa famille	Le patient et les membres de sa famille
Thérapies psychanalytiques	Le patient seulement	Le patient seulement
Thérapies comportementalistes	Le patient seulement	Le patient et possiblement les membres de sa famille

La thérapie familiale systémique a permis de développer le génogramme, un outil simple et efficace qui permet de recueillir, d'enregistrer et de visualiser les informations sur le patient et sa famille (Nguyen et Lambert, 1991; Béland, 1993). La section qui suit présente un aperçu détaillé du génogramme.

1.4.5 Description du génogramme

Le génogramme est une représentation graphique de la structure familiale d'un individu analogue à un arbre généalogique (fig. 1.1). Il est élaboré, à la main ou à l'aide d'un logiciel informatique, par le professionnel de la santé lors des séances de thérapie avec le patient ou les membres de sa famille. Il permet de visualiser et d'enregistrer les informations biomédicales et psychosociales de la famille sur deux ou trois générations (Béland, 1993).

Il est considéré comme un moyen de communication entre le clinicien et le patient et est souvent utilisé pour des fins thérapeutiques et de recherche clinique.

Un génogramme est essentiellement constitué de trois catégories d'informations concernant le patient et sa famille qui sont : la carte de la structure familiale, les informations sur la famille et les relations familiales (Gerson et McGoldrick, 1990 ; Bowen, 1978).

1.4.5.1 La carte de la structure familiale

Cette carte regroupe les différents liens biologiques et légaux qui unissent les membres de la famille sur au moins trois générations. Chaque lien est représenté par un symbole spécifique. Cependant, l'ensemble des symboles standardisé ne fait pas l'unanimité (Gerson et McGoldrick, 1990; Lambert, 2006).

1.4.5.2 Les informations sur la famille

Il s'agit des caractéristiques des membres de la famille du patient. Elles couvrent, entre autres, les renseignements démographiques (âge, sexe, etc.), les antécédents personnels (maladie, accident, etc.), l'éducation et le travail, les événements critiques et les expériences importantes (Béland, 1993). Ces informations sont représentées textuellement selon des règles qui ne font pas souvent l'unanimité.

1.4.5.3 Les relations affectives

Elles décrivent les relations (liens) affectives entre les différents membres de la famille. Selon les auteurs, ces relations sont classées dans quatre, six ou sept catégories (tabl. 1.2).

D'après Bowen (1978), il existe six types de relations affectives, soient :

- relation très proche et fusionnelle;
- relation fusionnelle et conflictuelle;
- relation proche;
- relation pauvre et conflictuelle;
- relation d'indifférence;
- relation distante.

Par ailleurs, Béland (1993) rajoute au génogramme une quatrième catégorie consacrée à l'histoire de la famille. Elle décrit les événements passés significatifs tels qu'un voyage ou un accident grave.

La description du génogramme nous révèle qu'il y'a certaines ambiguïtés et difficultés qui entourent la présentation des informations à cause du manque de standardisation des concepts et des symboles.

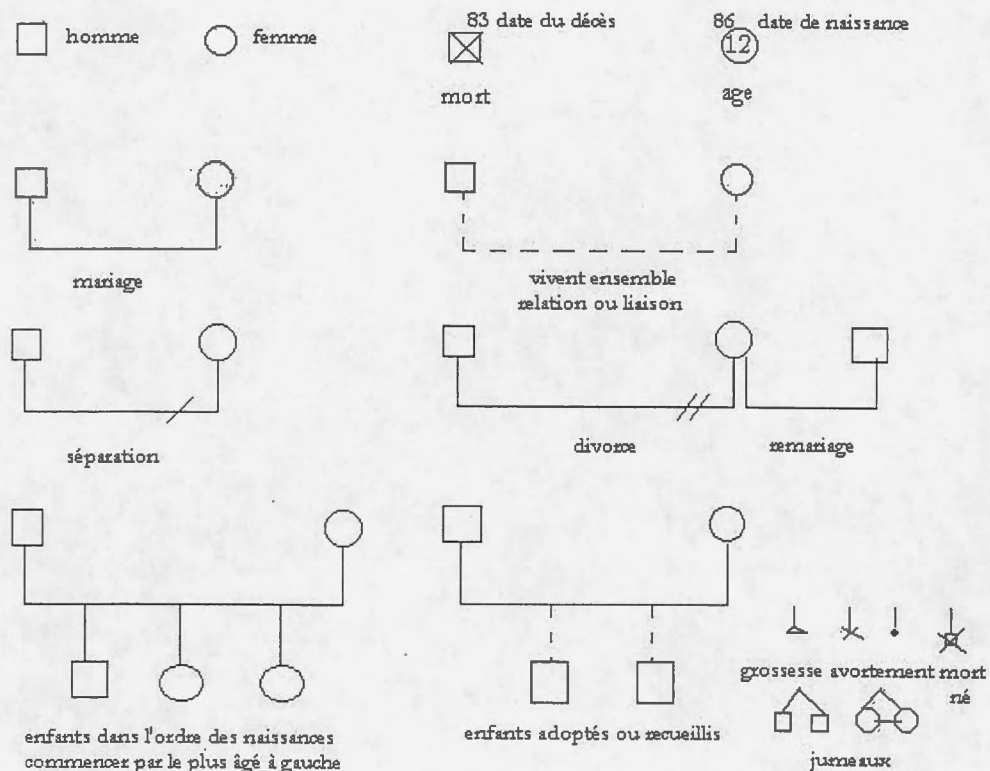


Figure 1.1 Exemple de symboles du génogramme

Nous avons présenté dans les sections précédentes le CJM et la thérapie familiale systémique. Ceci afin de comprendre le contexte et le terrain de notre recherche. Nous énoncerons dans le reste de ce chapitre la problématique et les questions de recherche ainsi que les objectifs de ce mémoire.

1.5 Problématiques

Les sections précédentes, basées sur la revue de la littérature et les entretiens avec le Dr. Lambert, nous ont permis d'identifier certaines problématiques rencontrées au CJM. Ces problématiques sont les suivantes :

- Les employés du CJM (médecins, psychologues et travailleurs sociaux, infirmiers, gestionnaires etc.) ont des formations différentes; donc ils n'utilisent pas tous le même vocabulaire professionnel. Ainsi, des problèmes de **synonymie** et de **polysémie** apparaissent. Par exemple, certains intervenants utilisent le terme « lien familial » tandis que d'autres utilisent le terme « relation familiale » pour décrire la structure familiale. Cette synonymie empêche, entre autres, de retrouver des informations pertinentes lors de la recherche par mots clés.
- **L'absence d'un document** (papier ou électronique) qui précise les règles d'utilisation des principaux termes du domaine, ce qui entraîne des interprétations différentes d'une même réalité d'une part. D'autre part, ce manque pousse les employés à faire plusieurs recherches souvent coûteuses en terme de temps.
- Les nouveaux employés rencontrent d'énormes difficultés à maîtriser le **jargon utilisé**. Ceci est dû à la multidisciplinarité des différents intervenants;
- Les nouveaux employés en particulier ont de la difficulté à **retrouver efficacement les informations** dont ils ont besoin à cause de leurs dispersions dans différents supports (bases de données, site Web, etc.).
- Dans le contexte de la thérapie familiale systémique, **les liens** (les relations) affectifs entre les membres de la famille ne sont pas classifiés de façon unanime. Certains auteurs les ont classifiés en quatre catégories, d'autre en six alors que d'autres en ont identifié sept (tabl. 1.2).
- les symboles du **génogramme** représentant les relations familiales ne font pas l'unanimité.

Ces problématique affectent négativement les employés dans les aspects suivant : la communication, la recherche d'informations et l'intégration. En effet, les problèmes de

synonymie et de polysémie entraînent des malentendus entre les personnes et entre les systèmes d'information. Par ailleurs, l'absence de document qui précise les règles d'utilisation des termes et leurs significations pousse les employés à faire des recherches dans plusieurs sources telles que les bases de données et les sites Internet .etc. Ces recherches sont souvent coûteuses en terme de temps d'une part, et les informations trouvées ne sont pas nécessairement pertinentes d'autre part.

Ces problématiques nous ont amené à poser les questions suivantes :

- Quelles sont les connaissances nécessaires (cruciales) que chaque employé devrait connaître et auxquelles il aurait accès rapidement et facilement afin d'éviter les confusions et de pouvoir communiquer (au sens large) efficacement avec ses collègues?
- Quelle serait la meilleure façon de représenter ces connaissances cruciales afin de les conserver et d'en faciliter l'accès?
- Comment pourrait on transférer ces connaissances et les partager entre les différents employés?

Tableau 1.2 : Classification des liens affectifs

Sept Liens affectifs (Genopro, 2006)	Six relations affectives (Bowen, 1978)	Quatre relations affectives (Béland, 1993)
Lien de proche	Proche	Rompue
Lien de fusion	Très proche et fusionnelle	Faible
Lien de très proche	Fusionnelle et conflictuelle	Conflictuelle
Lien de fusion-conflit	Distante	Fusionnelle
Lien de distance	Indifférente	
Lien d'éloignement	Pauvre et conflictuelle	
Lien de conflit		

1.6 L'objectif du mémoire

Notre objectif est d'appliquer une démarche de gestion des connaissances afin de résoudre les problèmes cités ci-dessus et répondre aux questions de recherche posées. Cette démarche est inspirée des projets de développement de systèmes de gestion des connaissances à base d'ontologies tels que le projet *Entreprise Ontology* (Uschold et al., 1998) et le projet *Gestion et Diffusion du Savoir en Télécommunication* (Lefebvre et al., 2003).

Cette démarche comporte la réalisation d'un système de gestion des connaissances basé sur une ontologie qui aide à assurer :

- une meilleure communication entre les différents intervenants du CJM. Ceci aide à améliorer leur performance et la qualité des services offerts aux jeunes;
- une optimisation du temps de recherche d'informations en mettant à la disposition des intervenants les informations et les connaissances dont ils ont besoins de façon simple, claire et bien structurée;
- un apprentissage et une intégration facile et rapide des nouveaux employés notamment les nouveaux diplômés.

Ce chapitre nous a permis de ressortir les problématiques reliées à la gestion des connaissances au sein du CJM. Cela a été réalisé grâce à une revue de littérature et des entretiens avec notre directeur, notre codirecteur comme nous allons le voir au chapitre suivant.

Dans ce chapitre nous avons aussi présenté le CJM, sa mission et ses services, en mettant en exergue la médecine familiale systémique, l'une des disciplines pratiquées au sein du CJM. Finalement, les objectifs du mémoire ont été énoncés. Ces objectifs seront atteints en suivant la méthodologie présentée au prochain chapitre.

CHAPITRE II

MÉTHODOLOGIE DE RECHERCHE

Comme mentionné dans le chapitre précédent, l'objectif de ce travail de recherche est d'appliquer une approche de GC au CJM afin de résoudre certains nombre de problématiques. Ces problématiques sont liées à la communication, la recherche d'informations, l'apprentissage et l'intégration des nouveaux employés.

Afin d'atteindre nos objectifs et de répondre aux questions de recherche posées, nous avons suivi une approche constituée de quatre principales phases. La première phase comporte trois étapes préalables qui nous ont aidé à bien définir notre projet de recherche et la problématique. La deuxième phase comporte les activités de recherche bibliographique et le recueil d'informations. La troisième phase consiste à analyser les données et à construire l'ontologie OCJM. Enfin la dernière phase sert à développer le prototype du système de gestion des connaissances du CJM.

2.1 La première phase : Définition du projet

C'est une phase préalable dont la finalité est de définir précisément notre projet de recherche et la problématique qui y est rattachée. Elle est constituée des trois étapes suivantes :

2.1.1 Identification du thème de la recherche

Le suivi de quelques cours de maîtrise en particulier le cours « fondements théoriques des systèmes d'information » et la lecture de certains articles nous ont convaincu de l'importance de la gestion des connaissances au sein des organisations. De plus, plusieurs auteurs soulignent l'absence d'une démarche unique et détaillée de GC dans les entreprises (Rubenstein-Montano et al., 2001). Ces deux éléments nous ont amené à cerner le thème de recherche.

Ce thème qui est la gestion des connaissances dans les organisations a été accepté par mon directeur de recherche et a été d'un grand intérêt au CJM grâce à mon codirecteur.

2.1.2 Revue de littérature et entretiens exploratoires

La seconde étape préalable a consisté à effectuer une revue de littérature consacrée à la gestion des connaissances et aux ontologies.

Parallèlement, des visites au CJM et des entretiens ont eu lieu avec mon codirecteur, ceci afin de comprendre de quelle manière les problèmes de gestion des connaissances pourraient être appréhendés.

2.1.3 Définition du projet de recherche

À l'issue de cette étape nous avons défini plus précisément notre projet de recherche. Il s'agit d'appliquer une démarche de gestion des connaissances supportée par un SGC basé sur une ontologie, afin de résoudre les problèmes cités au chapitre précédent.

Après ces trois étapes préalables, nous sommes passés à la deuxième phase consacrée à la revue de littérature et la collecte de données.

2.2 La deuxième phase : Revue de la littérature et collecte de données

Cette deuxième phase est constituée de deux étapes principales. La recherche bibliographique et les entretiens avec le Médecin en titre au CJM. Ces deux étapes n'étaient pas nécessairement disjointes, mais elles se chevauchaient dans plusieurs périodes.

2.2.1 La recherche bibliographique

Nous avons rencontré une difficulté à identifier les références bibliographiques pertinentes sur notre sujet de recherche. Cette difficulté a pour origine non pas le manque, mais l'abondance et la dispersion des recherches sur ce sujet. Cette dispersion est essentiellement due à la multidisciplinarité du sujet. Nous avons donc procédé à éliminer certains articles traitant des sujets tels que la GC et la comptabilité ou la GC et la finance.

La bibliographie que nous avons retenue concerne principalement la gestion des connaissances et les ontologies, en plus des documents sur la médecine familiale et les centres de jeunesse.

Au cours de cette étape, nous avons essayé de définir les principaux concepts (construits) et de faire une synthèse et une analyse comparative des principaux cadres conceptuels existants concernant la GC et les ontologies.

Cette approche synthétique et comparative nous a aidé d'une part à faire des choix éclairés pour réaliser notre projet de recherche et d'autre part à proposer un cadre conceptuel de base sur le rôle des ontologies (chap. 4).

2.2.2 Les entretiens avec le Médecin en titre au CJM

Cette étape nous a aidé principalement à identifier les problématiques liées à la GC citées au chapitre précédent.

Les entretiens que nous avons eus avec notre codirecteur et le personnel du CJM étaient plus souvent formels et quelques fois informels. Pour les entretiens formels, nous préparions d'avance des questions précises auxquelles nous voudrions avoir des réponses. Tandis que durant les entretiens informels nous posions des questions d'ordre général concernant le déroulement des activités au CJM ; la famille et les jeunes québécois. De plus, nous nous sommes échangés quelques informations complémentaires sur le sujet par téléphone et par courrier électronique.

Ces entretiens nous ont également permis d'identifier certaines exigences et spécifications pour le développement du prototype comme nous le verrons à la section (2.7).

Une fois la revue de littérature et la collecte de données achevées, nous sommes passés à la phase d'analyse de données et construction de l'ontologie OCJM tel que présenté à la section suivante.

2.3 La troisième phase : Analyse de données et construction de l'ontologie OCJM

La construction de l'ontologie est réalisée en suivant la méthodologie de Natalya et McGuiness (2003) inspirée de génie cognitif et présentée en détail avec d'autres méthodologies au chapitre quatre.

Nous pouvons synthétiser cette méthodologie, selon notre adaptation, en trois principales étapes. La première étape consiste à identifier les types de connaissances et d'informations

que le CJM devrait gérer pour résoudre les problématiques rencontrées et l'établissement d'une liste des principaux termes utilisés. Cette étape nous a permis d'identifier quatre domaines de connaissances que l'ontologie OCJM devrait contenir, à savoir : la médecine familiale, le génogramme, les personnes et le CJM (chap. 4). Ensuite nous avons dressé une liste de principaux termes désignant les concepts utilisés au CJM (tabl. 6.2 et appendice A).

La deuxième étape est la formalisation de l'ontologie et l'établissement d'une hiérarchie des classes (termes) (chap. 6 et appendices B et C).

La dernière étape consiste à l'évaluation et à la validation de l'ontologie par mon directeur et mon codirecteur.

La construction de l'ontologie OCJM est reprise en détail au chapitre six.

2.4 La quatrième phase : Développement du prototype du SGC

Le développement du prototype est réalisé en suivant les trois principales phases classiques du génie logiciel à savoir la phase d'analyse et de conception, la phase de réalisation ou de programmation et la phase de livraison.

2.4.1 Phase de conception

Dans cette phase nous avons listé un ensemble d'exigences et de spécifications identifiées lors des entretiens avec le personnel du CJM. Par exemple l'utilisation de la plateforme Windows au lieu d'Unix ou autre. Par la suite, nous avons élaboré l'architecture de l'application (fig. 7.2), choisi les serveurs d'application (Tomcat) et de base de données (MySQL). Nous avons également conçu les différents modules et choisi les langages de programmation (chap. 7).

2.4.2 Phase de réalisation

Dans cette phase nous avons commencé la réalisation des modules et l'écriture des programmes. Nous avons aussi créé la BD d'utilisateurs et la BD qui va contenir l'ontologie et les instances.

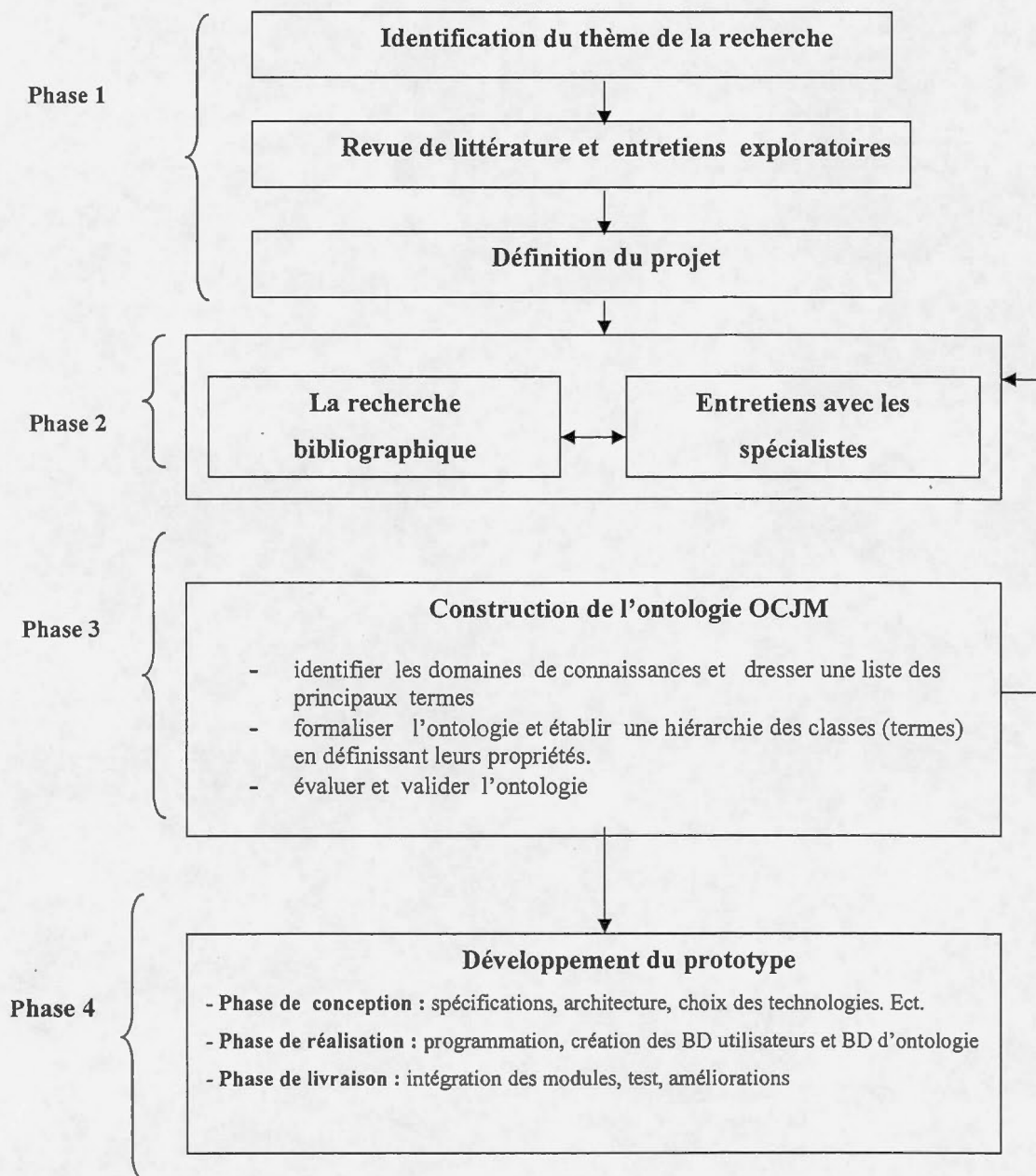
2.4.3 Phase de livraison

Dans cette dernière phase, nous avons intégré les différents modules (l'ontologie OCJM, BD, page JSP, modules JavaBeans) et testé la première version du prototype que nous avons amélioré par la suite.

Les étapes de cette méthodologie sont résumées dans la figure 2.1.

Le chapitre suivant expose l'état de l'art sur la gestion des connaissances dans les organisations.

Figure 2.1 : Approche méthodologique



CHAPITRE III

LA GESTION DES CONNAISSANCES

Depuis quelques années déjà, certains chercheurs de divers domaines ont constaté l'arrivée d'une nouvelle économie dominée par la production, l'échange et le partage de la connaissance (Zacklad et Grundstein, 2001).

Dans cette économie, la gestion des connaissances (GC, *knowledge management*) se taille une place de choix et constitue une clé importante qui permet aux organisations d'améliorer leur performance et d'obtenir un avantage concurrentiel (Ermine, 2003).

Selon un sondage de *American Management Association* réalisé en 1999, 79 % des PDG interrogés estiment que la gestion du savoir est vitale pour le succès de leur entreprise (Jacob et Pariat, 2000).

Par ailleurs, il est démontré que la GC est un processus qui peut être efficacement supporté par les ontologies comme nous le verrons dans ce chapitre et au chapitre suivant (Edgington et al., 2004).

Dans ce chapitre, nous présenterons d'abord l'état de l'art sur la gestion des connaissances des organisations, notamment les définitions et les cadres conceptuels. Par la suite, nous introduirons le rôle des ontologies dans la gestion des connaissances.

À la fin de ce chapitre, nous serons capable d'identifier les différents types de connaissances dans l'organisation et nous aurons une idée claire sur le processus de gestion des connaissances.

3.1 La connaissance dans l'entreprise

Nonaka (1994), l'un des fondateurs du « knowledge management », distingue les connaissances tacites des connaissances explicites. Les connaissances tacites sont personnelles, difficiles à formaliser et à communiquer, alors que les connaissances explicites sont codifiables et faciles à transmettre dans un langage formel. Par ailleurs et selon sa théorie sur la création des connaissances, il existe quatre modes de conversion des connaissances entre l'individu et le collectif ; entre le savoir tacite et le savoir explicite comme l'illustre la figure 3.1.

- **La socialisation** est l'échange des connaissances entre deux individus c'est à dire du tacite vers le tacite.
- **L'externalisation** est le transfert des connaissances tacites d'un individu, souvent expérimenté, vers un groupe d'individus (du tacite vers l'explicite).
- **L'intériorisation** est l'appropriation des connaissances organisationnelles par l'individu (de l'explicite vers le tacite).
- **La combinaison** est l'échange des connaissances d'un groupe d'employés vers l'ensemble de l'organisation. (de l'explicite vers l'explicite).

	Connaissances tacites	Connaissances explicites
Connaissances tacites	Socialisation	Externalisation
Connaissances explicites	Intériorisation	Combinaison

Figure 3.1 Les modes de conversion des connaissances (Nonaka, 1994)

3.2 La gestion des connaissances

Il existe dans la littérature de nombreuses définitions de la gestion des connaissances. Jacob et Pariat (2000) en ont présenté une douzaine. Cependant, selon la discipline et la tendance des auteurs, on peut regrouper ces définitions en quatre catégories (tabl.3.1) (Prax, 2003):

- **Définitions utilitaires :** selon le point de vue de l'utilisateur, la gestion des connaissances consiste à fournir, voire à diriger la bonne information à la bonne personne et au bon moment.
- **Définitions opérationnelles :** pour un gestionnaire des opérations, la gestion des connaissances est la combinaison des savoirs et savoir-faire dans les processus et dans les produits pour améliorer la performance de l'entreprise. Rossett (1999, cité par Jacob et Pariat, 2000) reconnaît que « *Knowledge management involves recognizing, documenting and distributing explicit and tacit knowledge in order to improve organizational performance* ».
- **Définitions économiques :** cette catégorie fait référence à la valorisation du savoir et du savoir-faire en tant qu'actif immatériel, fondement de la nouvelle économie du savoir. La gestion des connaissances dans ce contexte permet de qualifier ou même quantifier cet actif. Bouteillier (1999, cité par Jacob et Pariat, 2000) soutient que « La gestion des connaissances est une nouvelle science visant à réorganiser l'entreprise autour de sa richesse immatérielle ».
- **Définitions fonctionnelles :** les auteurs de ces définitions décrivent la gestion des connaissances à travers le cycle de vie de la connaissance. Les définitions de cette catégorie sont les plus répandues dans la littérature. Elles mettent l'accent sur les dispositifs qui seront mis en œuvre pour supporter le cycle de vie des connaissances. Pour Ermine (2001), la gestion des connaissances est un défi managérial s'inscrivant dans la vision stratégique globale de l'entreprise et elle vise à capitaliser, à créer et à partager l'ensemble des connaissances de l'organisation.

Tableau 3.1 : Les quatre catégories des définitions de la GC

Catégorie de définition	Description
Définitions utilitaires	La GC consiste à diriger la bonne information à la bonne personne et au bon moment.
Définitions opérationnelles	La GC est la combinaison des savoirs et savoir-faire dans les processus et dans les produits pour améliorer la performance de l'entreprise
Définitions économiques	La GC est une nouvelle science visant à réorganiser l'entreprise autour de sa richesse immatérielle
Définitions fonctionnelles	La GC vise à capitaliser, à créer et à partager l'ensemble des connaissances de l'organisation.

Par ailleurs, la définition proposée par Grundstein (2003) et consolidée par de nombreux chercheurs (Zacklad et Grundstein, 2001), nous semble la plus complète et la plus pertinente.

Selon Grundstein, la gestion des connaissances est le « Management des activités et des processus destinés à amplifier l'utilisation et la création des connaissances au sein d'une organisation selon deux finalités complémentaires fortement intriquées : une finalité patrimoniale et une finalité d'innovation durable ; finalités sous-tendues par leurs dimensions économiques et stratégiques, organisationnelles, socioculturelles et technologiques».

La finalité patrimoniale, pose des problèmes liés aux connaissances explicites, comment les repérer, les préserver, les valoriser et les actualiser; tandis que la finalité d'innovation durable pose des problèmes liés à l'apprentissage organisationnel.

Cette définition a amené Grundstein à concevoir un cadre conceptuel (cycle de vie) pour la gestion des connaissances. Ce cadre est constitué d'un ensemble de problématiques à cinq facettes (fig. 3.2). Il s'agit de :

1. La facette **repérer** : elle consiste à identifier les connaissances cruciales de l'organisation, à les localiser et à les hiérarchiser.
2. La facette **préserver** : une fois les connaissances cruciales sont repérées, il faut les modéliser, les formaliser et les conserver. En d'autres termes, il faut les expliciter formellement et les sauvegarder. Cependant, lorsque les connaissances ne sont pas explicites, il faut encourager les processus de socialisation et de formalisation.
3. La facette **valoriser** : afin de mettre les connaissances au service du développement et de l'expansion de l'entreprise, il faut les rendre accessibles selon certaines conditions, les diffuser, les partager, les exploiter, les combiner et créer des connaissances nouvelles.
4. La facette **actualiser** : les connaissances devraient être évaluées et mises à jour régulièrement. De plus, il faut les standardiser et les enrichir par d'autres connaissances.
5. La facette **manager** : elle concerne les interactions entre les différentes facettes mentionnées précédemment.

Par ailleurs, Grundstein (2003) a remarqué que le concept de la gestion des connaissances est influencé par trois principaux courants : le courant économique et managérial, le courant intelligence artificielle et ingénierie des connaissances et enfin le courant ingénierie des systèmes d'information.

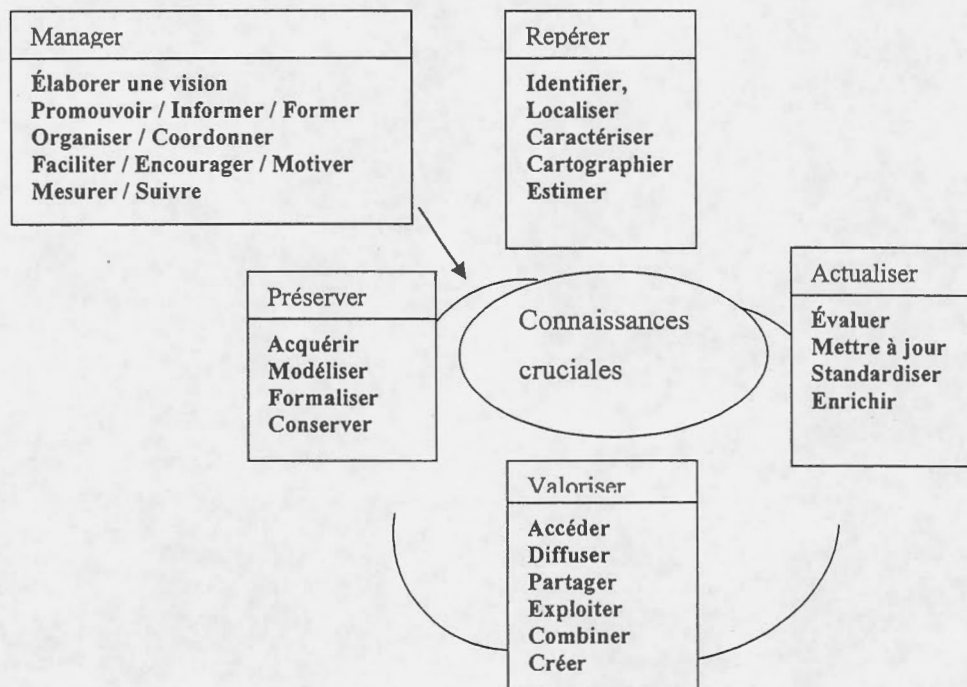


Figure 3.2 : Cadre conceptuel de la GC (Grundstein, 2003)

3.3 Mémoire de l'entreprise

La mémoire de l'entreprise est l'ensemble des connaissances d'une entreprise ou d'une entité de l'entreprise. Elle est construite pour aider les employés à mieux exploiter leur patrimoine de connaissance. Elle inclut aussi bien la mémoire technique (constituée des connaissances techniques) que la mémoire managériale (constituée des connaissances organisationnelles). Ces connaissances peuvent se retrouver dans des documents de types variés, dans l'intellect des employés, dans des bases de données ou dans des bases de connaissances (Gaëlle, 2001). Tourtier (1995, cité par Dieng et al., 1998) a distingué quatre types de mémoires d'entreprise, selon leur contenu et leur objectif (tabl. 3.2):

- la mémoire *métier* comporte les connaissances liées à un métier donné. Elle peut décrire les documents, les outils et les méthodes employés dans le métier en question;
- la mémoire *société* décrit la structure de l'organisation, ses activités, ses produits et ses partenaires (clients, fournisseurs, sous-traitants de l'entreprise, etc.);
- la mémoire *individuelle* précise les informations concernant un membre donné de l'entreprise telles que son statut, ses compétences, et son expérience;
- la mémoire *de projet* comporte la définition du projet ainsi que les différentes phases de son déroulement.

D'autre part, on peut aussi distinguer les mémoires de l'entreprise selon le mode et la portée de leur exploitation (Gaëlle, 2001).

Tableau 3.2 : Les quatre types de mémoire d'entreprise (1995, cité par Dieng et al.,1998)

Type de mémoire	Description
Mémoire <i>métier</i>	- connaissances concernant les métiers
Mémoire <i>société</i>	- connaissances concernant l'organisation
Mémoire <i>individuelle</i>	- connaissances concernant les employés
Mémoire <i>de projet</i>	-connaissances concernant les projets

3.4 Systèmes de gestion des connaissances

La définition du système de gestion des connaissances (SGC) est encore peu précise et peu claire. Ceci est dû à la nouveauté du domaine de gestion des connaissances et à son interdisciplinarité (Ouni et Dudezert, 2004). Plusieurs auteurs ont identifié trois grandes approches de définition du SGC (Ouni et Dudezert, 2004; Alavi et Leidner, 1999), il s'agit de :

- L'approche **technologie de l'information-Système d'information (TI-SI)** : cette approche s'intéresse à la dimension technologique du système nécessaire pour le soutien des processus de GC.
- L'approche **tactique** : cette catégorie met l'emphasis sur les aspects organisationnel et culturel de la GC pour la réalisation des objectifs stratégiques de l'entreprise.
- L'approche **systémique** : cette approche adopte une vision globale du SGC dans son environnement. Le SGC est constitué d'un ensemble de sous-systèmes en interaction. Toutes les dimensions technologique, organisationnelle, culturelle, et stratégique sont prises en considération.

Selon Gaëlle (2001), un système de gestion (capitalisation) des connaissances est une infrastructure informatique permettant la mise en œuvre de la mémoire technique de l'entreprise, son enrichissement, sa mise à jour ainsi que son utilisation. Le SGC permet en particulier la diffusion et le partage de ce patrimoine de connaissances. Nous optons pour cette définition appartenant aux approches SI/TI citées ci-dessus.

3.5 Les ontologies et la gestion des connaissances

L'échange et le partage des connaissances au sein d'une organisation passe nécessairement par une communication entre personnes et systèmes d'information (SI).

Cependant, du fait de leur formation et de leurs expériences différentes, les employés d'une même entreprise utilisent un vocabulaire et des jargons différents. Ils peuvent aussi émettre des points de vue divergents sur un même sujet. Ceci entraîne une mauvaise communication et par conséquent une mauvaise gestion des connaissances (Dieng et al., 1998).

De plus, la communication entre les systèmes d'information est souvent confrontée au problème d'interopérabilité dû à l'hétérogénéité, schématique ou sémantique, des sources de données.

Par ailleurs, les organisations et les employés ont des besoins énormes en recherche d'informations localement ou sur Internet et ils sont souvent noyés dans une foule d'informations non pertinentes.

Afin de résoudre ce type de problématique, les ontologies en tant qu'« une représentation explicite d'une compréhension partagée » (Gruber, 1993) aident à améliorer la communication entre personnes, aident à résoudre les problèmes d'interopérabilité, facilitent et améliorent la recherche d'informations.

Le prochain chapitre reprend en détail la définition des ontologies et présente un cadre conceptuel de base sur le rôle des ontologies dans la gestion des connaissances.

CHAPITRE IV

LES ONTOLOGIES

Nous avons clos le chapitre précédent par une brève présentation du rôle des ontologies dans la gestion des connaissances des organisations. Dans ce chapitre, nous exposerons en détail les ontologies en décrivant notamment leurs définitions, leurs composantes, leur typologie, leurs rôles et finalement leurs approches de développement. Ce chapitre est constitué de plusieurs études comparatives qui permettent d'avoir une vue synthétique, complète et détaillée sur les ontologies.

4.1 Définitions

Le terme 'ontologie' est emprunté à la philosophie où il désigne la « Partie de la métaphysique qui s'applique à « l'être en tant qu'être », indépendamment de ses déterminations particulières. » (OQLF, 2005). Elle s'intéresse à l'ordre et la structure de la réalité (Holsapple et al., 2002).

Au fil des années, plusieurs définitions d'ontologies sont proposées par différentes communautés scientifiques. Cependant, un nombre important de chercheurs s'accorde sur celle de Grüber (1993). Selon Grüber une ontologie est une « spécification explicite d'une conceptualisation ».

Dans cette section nous présenterons les principales définitions citées dans la littérature et nous expliquerons les éventuelles relations qui existent entre elles.

Les principales définitions citées dans la littérature pourraient être regroupées en trois principaux ensembles : les définitions descriptives, les définitions basées sur la logique et les définitions prescriptives (tabl. 4.1).

4.1.1 Définitions descriptives

Cet ensemble comporte les définitions qui mettent l'emphasis sur les caractéristiques des ontologies.

En 1991, Neches et al. ont défini une ontologie comme suit: « *An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary* ».

Selon cette définition, une ontologie est un ensemble de termes et de relations entre ces termes. Elle inclut aussi des règles d'inférence entre ces termes et ces relations.

En 1993, Grüber propose sa célèbre définition qui décrit une ontologie comme une « spécification explicite d'une conceptualisation ».

Cette définition est la plus citée dans la littérature et constitue la base de plusieurs autres définitions proposées par la suite.

En 1997, Borst a raffiné la définition de Grüber de la manière suivante : « une ontologie est une spécification formelle d'une conceptualisation partagée ».

En 1998, Studer et al. ont fusionné les deux définitions précédentes :

« Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée. »
 Une 'conceptualisation' fait référence à un modèle abstrait de la réalité d'un domaine donné. Une 'spécification explicite' précise que les concepts et les relations dans le modèle abstrait doivent avoir des noms et des définitions explicites. Le terme 'formelle' indique que la spécification doit être encodée dans un langage bien compris tels que les langages basés logique. Finalement, le mot 'partagée' signifie que l'ontologie doit capturer des connaissances consensuelles et devrait être utilisée par différentes applications ou personnes (Uschold et Gruninger, 2004).

4.1.2 Définitions basées sur la logique

Les principaux auteurs de ces définitions sont Guarino et Giaretta (1995). Ils considèrent une ontologie comme « *a logical theory which gives an explicit, partial account of a conceptualization* ».

Ces auteurs ont amélioré leur définition en 1998 et ils ont annoncé qu'une ontologie est « *a set of logical axioms designed to account for the intended meaning of a vocabulary*. »

4.1.3 Définitions prescriptives

Les définitions de cette catégorie sont fondées sur les méthodologies et les processus de construction d'ontologies. Pour Bernaras et al. (1996, cité par Gomez et Benjamins, 1999) *"An ontology provides the means for describing explicitly the conceptualization behind the knowledge represented in a knowledge base"*. De même, Swartout et al. (1997 cité par Mendes et al., 2003) proposent la définition suivante: *"An ontology is a hierarchically structured set of terms for describing a domain that can be used as a skeletal foundation for a knowledge base "*.

La lecture attentive de ces définitions nous permet de déduire que ces dernières sont complémentaires. Elles sont toutes importantes, car chacune d'elle met l'accent sur une dimension particulière des ontologies. L'ensemble de ces définitions nous a permis de comprendre clairement ce que sont une ontologie et ses caractéristiques.

4.2 Les composantes d'une ontologie

La conception et la construction des ontologies se font en général selon deux principales approches : l'approche des cadres (*frames*) et la logique du premier ordre ; et l'approche des logiques de description (*description logics*). La première approche permet de formaliser les connaissances en utilisant les cinq éléments suivants : les classes, les relations, les fonctions, les axiomes formels et les instances. La deuxième approche permet de véhiculer les connaissances à l'aide des trois types de composantes suivantes : les concepts, les attributs et le individu.

4.2.1 Modélisation des ontologies par les cadres et la logique du premier ordre

Selon Grüber (1993), une ontologie construite à l'aide des techniques de cadres et de la logique de premier ordre est constituée de cinq principales composantes :

Tableau 4.1: Les trois catégories de définitions d'ontologies

Catégories	Définitions /construits	Auteurs
Descriptive (caractéristiques des ontologies)	<ul style="list-style-type: none"> - termes, relations, règles d'inférence - spécification explicite, conceptualisation - spécification formelle, conceptualisation partagée - spécification formelle, explicite conceptualisation partagée. 	<p>Neches et al. (1991)</p> <p>Grüber (1993)</p> <p>Borst (1997)</p> <p>Studer et al. (1998)</p>
Prescriptives (méthodologies et processus pour construire des ontologies)	<ul style="list-style-type: none"> - les moyens pour décrire la conceptualisation - base de connaissances 	<p>Bernaras et al. (1996)</p> <p>Swartout et al. (1997)</p>
Basées sur la logique	<ul style="list-style-type: none"> - théorie logique - axiomes logiques 	Guarino et Giaretta (1995)

- a. **Les classes** : elles représentent les concepts généraux d'un domaine donné. Une classe pourrait représenter une tâche, un processus, etc. Les classes sont organisées en taxonomie hiérarchique.
- b. **Les relations** : elles représentent des associations existant entre les concepts du domaine. Habituellement, les ontologies contiennent des relations binaires. Le premier argument d'une relation est le domaine de la relation, tandis que le second argument est son image. Des exemples de relations incluent : des relations de généralisation et de spécialisation « **sous-classe** »; des relations d'agrégation ou de composition « **fait partie de** ». Formellement, une relation est définie comme un sous-ensemble de produit de n éléments : $R \subset C_1 * C_2 * \dots * C_n$.
- c. **Les fonctions** : ce sont des cas particuliers de relations. Une fonction représente une relation entre n éléments dont le n ème élément est unique pour les $n-1$ éléments précédents. Une fonction est formellement définie par :

$$F : C_1 * C_2 * \dots * C_{n-1} \longrightarrow C_n$$
- d. **Les axiomes formels** : ils permettent de représenter les assertions qui sont toujours vraies. Ils sont utilisés pour inférer de nouvelles connaissances.
- e. **Les instances** : elles servent à représenter les éléments ou les individus des classes de l'ontologie.

4.2.2 Modélisation des ontologies par la logique de description

La théorie de la logique de description permet de construire les ontologies à l'aide de trois composantes : les concepts, les attributs (*rôles*) et les individus.

- **les concepts** : ils ont le même sens qu'en paradigme de cadres, ils représentent les classes d'objets.
- **les attributs** : ils permettent de décrire les relations binaires entre les concepts ainsi que les propriétés des concepts.
- **les individus** : ils représentent les instances des classes.

Par ailleurs et d'après (Gómez et Benjamins, 1999), il existe deux autres approches de modélisation d'ontologies, à savoir : l'approche de « software engineering » en utilisant UML et l'approche utilisée par la communauté des bases de données.

Il apparaît ainsi qu'une ontologie est plus complexe qu'un thésaurus ou une taxonomie (fig. 4.1). Un thésaurus est un ensemble de termes ayant entre eux des relations sémantiques et génériques. Il est orienté vers **l'information syntaxique**. Une taxonomie est une classification hiérarchique du vocabulaire contrôlé, elle est orientée vers **l'information sémantique**. Une ontologie contient une hiérarchie de termes mais de plus, elle explicite les relations logiques qui existent entre ces termes à l'aide d'un langage formel. Elle est orientée vers **la communication** (Edgington et al., 2004).

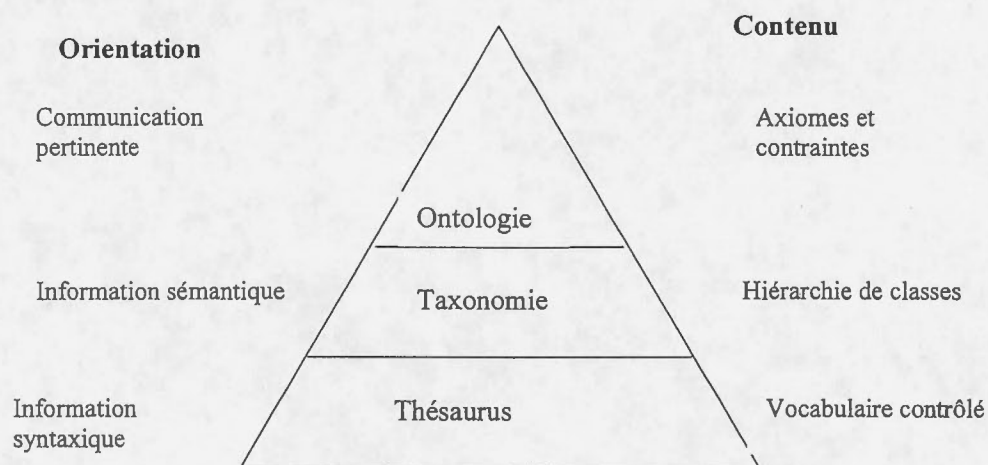


Figure 4.1 : Thésaurus, taxonomie et ontologie (Edgington et al., 2004).

4.3 Typologie d'ontologies

La classification des ontologies a fait l'objet de nombreuses recherches. Parmi celle-ci, nous en présenterons les deux plus représentatives. La première, proposée par Heijst et al. (1997), classe les ontologies selon leur objet de conceptualisation tandis que la deuxième,

proposée par Uschold et Jasper (1999), les classifient selon leur degré de formalité de représentation.

4.3.1 Classification selon l'objet de conceptualisation

Heijst et al. (1997), ont distingué quatre types d'ontologies selon leur objet de conceptualisation :

- **Les ontologies *du domaine*** décrivent les concepts spécifiques à un domaine. Par exemple, en médecine familiale, une ontologie du domaine décrira les concepts union, parenté, alcoolisme. etc.
- **Les ontologies *applicatives*** expriment des connaissances d'un domaine donné nécessaires à une application spécifique. Par exemple, en médecine familiale une ontologie applicative sera consacrée aux concepts utilisés pour les génogrammes.
- **Les ontologies *génériques*** contiennent des concepts abstraits de haut niveau valables pour plusieurs domaines. Par exemple, les concepts de temps et de l'espace peuvent être représentés dans une ontologie générique.
- **Les ontologies *de représentation*** conceptualisent les primitives des langages de représentation des connaissances. Par exemple, une ontologie sur le formalisme des graphes conceptuels.

4.3.2 Classification selon le degré de formalisation

Par rapport au degré de formalité de représentation, Uschold et Jasper (1999) proposent une classification comprenant quatre types d'ontologie :

- a. **ontologie informelle** : l'ontologie est exprimée dans un langage naturel ;
- b. **ontologie semi-informelle** : elle est exprimée dans un langage naturel structuré ;
- c. **ontologie semi-formelle** : elle est exprimée dans un langage artificiel défini formellement ;
- d. **ontologie formelle** : elle est exprimée dans un langage artificiel contenant une sémantique formelle.

Uschold et Jasper (1997) soulignent qu'une ontologie peut être informelle si elle est conçue uniquement pour la communication entre personnes, à condition qu'elle soit claire et précise. Cependant, si l'ontologie est conçue pour être employée par les systèmes informatiques alors elle doit être formelle.

Après la comparaison et l'analyse des différentes définitions, des typologie et des approches de modélisation d'ontologies, nous avons constaté que :

- une ontologie contient nécessairement un vocabulaire des termes désignant les « choses » du domaine concerné et des spécifications de leurs significations;
- elle inclut aussi des relations entre ces termes;
- elle peut avoir plusieurs formes et différentes composantes;
- elle doit être exprimée dans un langage précis.

Ces caractéristiques permettent de bien structurer les connaissances d'un domaine donné et de restreindre l'interprétation des termes afin de faciliter la communication au sein de l'organisation.

La suite de ce chapitre présente les principaux rôles des ontologies et les principales approches de développement.

4.4 Rôles des ontologies

Les ontologies sont utilisées dans des domaines et des contextes extrêmement variés ; ceci explique le manque d'un cadre conceptuel qui représente les différents rôles des ontologies. Par ailleurs, plusieurs auteurs ont souligné un ou plusieurs rôles des ontologies dans un contexte précis. Nous avons répertorié les principaux rôles des ontologies et nous avons conçu une base de cadre conceptuel à enrichir. Ce cadre est constitué des six facettes suivantes (fig 4.2 et tabl. 4.2) :

- La représentation des connaissances et le raisonnement sur ces connaissances.
- Le partage et la réutilisation des connaissances.
- La communication entre personnes.

- La recherche d'informations.
- L'intégration sémantique des sources de données.
- L'interopérabilité entre les systèmes d'information.

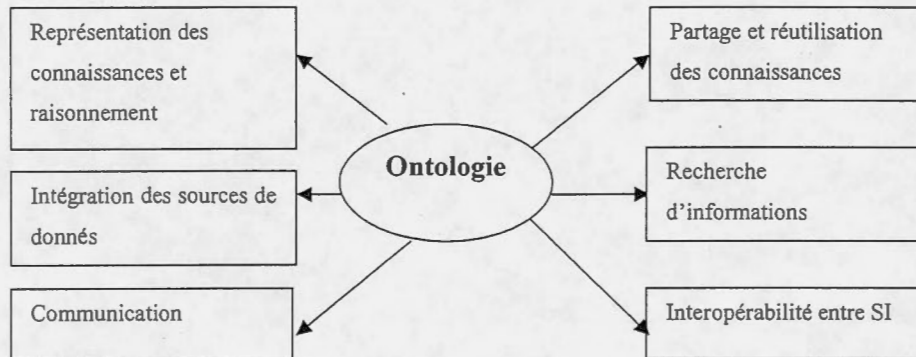


Figure 4.2 : Les rôles des ontologies

4.5 Approches de développement des ontologies

Guarino et Giaretta (1995) ont remarqué que le domaine d'ontologie évoluait vers une ingénierie ontologique. Cette ingénierie s'intéresse aux aspects pratiques (les méthodologies et les outils) qui permettent d'appliquer en pratique les résultats théoriques. Par ailleurs, Mendes et al. (2003) affirment qu'il n'existe pas encore de consensus à propos des méthodologies et des normes à adopter lors du processus de construction des ontologies. De leur côté, Natalya et McGuinness (2003) affirment qu'il n'y a pas qu'une seule méthodologie correcte de conception d'ontologies. Mendes et al. (2003) ont même dénombré un total de trente trois méthodologies de développement d'ontologies.

Tableau 4.2 : Les rôles des ontologies

Rôles	Contexte/problématiques	Objectifs	Auteurs
Représentation des connaissances et raisonnement	expliciter les connaissances tacites - règles d'utilisation des termes	- conservation des connaissances à long terme	Gruber, 1995 ; Sowa, 2000; Uschold et Gruninger, 2004 Natalya, 2004
Partage et réutilisation des connaissances	-multidisciplinarité - travail de collaboration	- réduire les coûts	Gruber, 1995; Edgington et al., 2004 ; Natalya , 2004
Communication	-multidisciplinarité - apprentissage	- améliorer la performance de l'organisation	Edgington et al., 2004
Recherche d'informations	- recherche sur le Web - recherche dans une base de connaissances	- améliorer la précision - réduire le temps	Uschold et al., 2004 ; Nenad et al., 2003
Intégration des sources de données	- sources de données hétérogènes - schéma global	- accès et interrogation de multiples sources hétérogènes	Uschold et al., 04 ; Wache et al., 2001 ; Obrst, 2003
Interopérabilité entre SI	- SI hétérogène	- collaboration entre différents systèmes	Gruber, 1995 ; Uschold et al., 2004 Obrst, 2003

Par ailleurs, dans une étude comparative, Fernández (1999) a comparé cinq principales méthodologies représentatives selon neuf critères. Le résultat de cette étude est présenté dans le tableau ci-dessous (tabl. 4.3). Après l'analyse des données de ce tableau, Fernandez arrive aux conclusions suivantes :

- METHONTOLOGY est la plus mature des cinq méthodologies;
- SENSUS est la seule méthodologie qui a une approche complètement différente des cinq autres méthodologies;
- Ces méthodologies constituent un point de départ pour développer une ou plusieurs méthodologies standards et adaptées aux différents types d'ontologies.

De même, Wache et al. (2001) ont comparé les principales méthodologies existantes. Ils ont affirmé que la méthodologie de Uschold et al. (1998) pourrait constituer un bon standard car elle est basée sur des phases indépendantes du domaine d'application d'ontologies.

Par ailleurs, Natalya et McGuinness (2003) en se basant sur leurs expériences, ont proposé une méthodologie spécifique pour la construction des ontologies de domaine. Cette méthodologie de génie cognitif est largement inspirée des principales méthodologies existantes comme nous le verrons dans les sections suivantes.

Dans ce qui suit, nous présenterons en détail la méthodologie de Uschold et al., la méthodologie METHONTOLOGY et finalement celle de Natalya et McGuinness. Cette dernière sera suivie pour développer l'ontologie du centre de jeunesse de la Montérégie (OCJM).

Tableau 4.3 : Comparaison des principales méthodologies d'ontologies
Fernandez, 1999 (traduit par Sogbohossou, 2002)

Méthodologies Critères	Uschold et King	Gruniger et Fox	Bernaras et al.	METHONTOLOGY	SENSUS
Héritage du génie de la connaissance	partiel	faible	fort	fort	aucun
Détail de la méthodologie	très peu	peu	très peu	beaucoup	moyen
Formalismes recommandés	aucun en particulier	logique	aucun	aucun	réseaux sémantiques
Stratégies de conception des applications	indépendante	semi- dépendante	dépendante	indépendante	semi- dépendante
Stratégies pour les concepts	centrifuge	centrifuge	descendante	centrifuge	non spécifiée
Cycle de vie recommandé	aucun	à détailler	aucun	prototype en evolution	à détailler
Différence avec la norme IEEE 1074- 1995	processus et activités manquants	processus et activités manquantes	processus et activités manquantes	processus de pré- développement et activités manquants	processus et activités manquantes
Techniques recommandées	inconnues	inconnues	inconnues	quelques activités manquantes	inconnues
Ontologies et applications	un domaine seulement	un domaine seulement	un domaine seulement	plusieurs domaines	plusieurs domaines

4.5.1 La méthodologie de Uschold et al.

La méthodologie de Uschold et al. (1998) est fondée sur l'expérience du développement de « *Entreprise Ontology* », un important projet d'ontologie pour les processus d'affaires d'entreprise. Il a été réalisé par l'institut de l'intelligence artificielle de l'université d'Edinburgh en partenariat avec IBM et d'autres firmes (Uschold et al., 1998).

Cette méthodologie est constituée des quatre principales phases suivantes:

1. Identifier les objectifs et la portée de l'ontologie

Dans cette phase, on doit d'abord répondre à la question « pourquoi l'ontologie est-elle construite et quelles sont les utilisations envisagées ? ». Ensuite, il faut identifier les utilisateurs de l'ontologie et leurs caractéristiques.

2. Construire l'ontologie

La construction de l'ontologie comporte les activités suivantes :

a. capturer l'ontologie :

- identifier les concepts et les relations clés du domaine;
- identifier les termes par lesquels on fait référence à ces concepts et relations. Par exemple « Service » et « Travail social » sont des termes utilisés au CJM et l'affirmation « le diabète est une maladie chronique » constitue une relation;
- donner des définitions textuelles non ambiguës pour ces concepts et relations;
- être d'accord (les experts du domaine) sur ces concepts, ces relations et leur définition.

b. coder l'ontologie : structurer et représenter les connaissances du domaine dans un modèle conceptuel en choisissant un langage de représentation adéquat.

c. intégrer les ontologies existantes : décider s'il est préférable d'utiliser les ontologies existantes pour accélérer le processus de développement.

3. **Évaluer l'ontologie** : vérifier et valider l'ontologie. Pour cela, on peut adapter les méthodes adoptées pour les systèmes à base de connaissances.
4. **Documenter l'ontologie** : en particulier toutes les hypothèses importantes et les primitives de base utilisées (classe, entité, etc.).

4.5.2 La méthodologie METHONTOLOGY

Cette méthodologie a été développée au laboratoire d'intelligence artificielle de l'université polytechnique de Madrid. Le cadre conceptuel METHONTOLOGY permet de construire des ontologies au niveau des connaissances (Fernandez, 1999). Il inclut trois principaux volets : l'identification du processus de développement de l'ontologie, un cycle de vie basé sur des prototypes évolutifs et des techniques spécifiques pour réaliser chaque activité.

1. Processus de développement de l'ontologie

Il comporte les activités suivantes :

- a. **Activités de gestion de projet** : planification, contrôle et assurance qualité.
- b. **Activités de développement** : spécification, conceptualisation, formalisation, implémentation et maintenance.
- c. **Activités de support** : acquisition de connaissances, évaluation, intégration d'autres ontologies, documentation et gestions des configurations.

2. Un cycle de vie basé sur des prototypes évolutifs : il s'agit d'identifier les phases de l'évolution de l'ontologie.

3. Des techniques spécifiques pour réaliser chaque activité

Il s'agit de :

- a. spécifier les étapes à suivre pour réaliser chaque activité;
- b. spécifier les techniques à utiliser;
- c. spécifier la façon d'évaluer les prototypes.

4.5.3 La méthodologie de Natalya et McGuinness

Natalya et McGuinness (2003) ont proposé une simple méthodologie de génie cognitif. C'est une approche itérative inspirée des méthodologies existantes telle que celle de Uschold et al. (1998). Elle est constituée des étapes suivantes :

- 1) définir le domaine et la portée de l'ontologie. Ceci pourra se faire en répondant à quelques questions de compétences;
- 2) intégrer d'autres ontologies existantes (optionnel);
- 3) établir une liste exhaustive des termes importants du domaine;
- 4) définir les classes et la hiérarchie des classes. L'hiérarchie peut se faire de trois façons distinctes : de haut en bas, de bas en haut ou en combinant les deux façons;
- 5) définir les propriétés (les attributs) des classes;
- 6) définir les facettes des propriétés;
- 7) créer les instances des classes;
- 8) évaluer et tester l'ontologie (applications, experts).

Nous avons choisi la méthodologie de Natalya et McGuinness (2003) pour développer l'ontologie OCJM pour les raisons suivantes :

- la simplicité de cette méthode convient à notre ontologie qui est relativement de petite taille;
- elle est détaillée dans un guide de création d'ontologies;
- elle est indépendante du domaine d'application;
- elle ne requiert aucun formalisme de représentation spécifique.

Dans ce chapitre, nous avons présenté et analysé les différentes définitions d'ontologies. De même, nous avons décrit leurs principaux rôles sous forme d'une base de cadre conceptuel. Finalement, nous avons exposé et comparé leurs principales approches de modélisation et de développement.

Ce chapitre nous a permis de choisir de façon éclairée la méthodologie de Natalya et McGuinness (2003) pour construire l'ontologie OCJM.

Dans le chapitre qui suit, nous verrons les différents langages de représentation d'ontologies, ce qui nous permettra de choisir le langage adéquat pour coder l'ontologie OCJM.

CHAPITRE V

LANGAGES D'ONTOLOGIES

Pour qu'une ontologie soit exploitée et intégrée dans un système d'information, elle doit être exprimée dans un langage formel permettant de traduire les composantes de l'ontologie et d'effectuer des inférences sur les connaissances représentées par ces composantes.

La littérature est riche en langages de représentation de connaissances dédiés ou adaptés aux ontologies. En général, ces langages sont basés sur un ou deux des paradigmes de représentation de connaissances suivants : le paradigme de la logique du premier ordre, le paradigme des graphes conceptuels, le paradigme des logiques de description « *description logic* (DL) » et le paradigme de cadres « *Frames* ».

D'autre part, ces langages sont regroupés en langages traditionnels et langages Web comme nous le verrons plus loin dans ce chapitre.

Ce chapitre présente les quatre paradigmes précités et les principaux langages sous-jacents. Il met en particulier l'accent sur les langages Web, notamment sur les derniers standards de l'organisation W3C, RDF et OWL.

5.1 Les paradigmes de représentation des connaissances

5.1.1 Le paradigme de la logique du premier ordre

C'est une logique formelle dont la grammaire est suffisamment riche pour formaliser un grand nombre de raisonnements logiques. La logique de premier ordre permet de représenter (à l'aide de constructeurs) des concepts, des fonctions, des prédicats (propriétés) ainsi que des connecteurs logiques. Elle dispose aussi de deux symboles \forall et \exists appelés quantificateurs

permettant de formaliser des énoncés tels que : «Tout X est Y» et «Il existe un X tel que pour tout Y, X RY».

Les langages de représentations de connaissances combinent souvent la logique de premier ordre avec l'un des autres paradigmes ci-après.

5.1.2 Le paradigme des cadres « *frames* »

Dans le contexte des langages d'ontologies, un cadre est un objet nommé, utilisé pour représenter un certain concept dans un domaine donné. Un cadre représente n'importe quelle primitive conceptuelle dotée d'attributs (slots) pouvant prendre différentes valeurs (facettes) (Kifer et al., 1995). Une autre façon de présenter ces attributs consiste à les considérer comme des relations binaires entre les concepts (classes). Le premier argument de cette relation est appelé domaine et le second est l'image (*range*). Dans ce paradigme on peut aussi utiliser des instances de classes et des fonctions qui sont des cas particuliers de relations. Une fonction lie un ensemble de classes à une valeur calculée à partir des valeurs d'attributs de classes. Ce paradigme fait souvent recours à la logique de premier ordre tel que l'utilisation des quantificateurs \forall et \exists . Les langages F-Logic et Ontolingua sont les exemples les plus connus qui implémentent ce paradigme.

Nous remarquons qu'il y a une correspondance entre le paradigme de cadres et l'approche orientée objet où les classes et les instances correspondent avec les cadres, les attributs et les associations de classes avec les slots. Entre les cadres, il y a aussi la spécialisation qui donne l'héritage dans les concepts de l'approche orientée objet.

5.1.3 Le paradigme de la logique de description

C'est un formalisme de représentation de connaissances basé sur la logique de premier ordre. La DL combine des représentations intentionnelles (TBox) et des représentations extensionnelles (ABox). Une Tbox (T pour Terminology) contient les définitions intentionnelles des connaissances organisées en concepts et rôles (relations) entre les concepts comme dans le cas des cadres. Une ABox (A pour Assertion) contient les

déclarations d'individus qui sont des instances des concepts définis dans la Tbox. La DL utilise largement les constructeurs.

5.1.4 Le paradigme des graphes conceptuels

Le paradigme des graphes conceptuels appartient à la famille des réseaux sémantiques. Un réseau sémantique représente les connaissances sous forme de graphes dont les nœuds représentent les concepts et les arêtes représentent les relations entre les concepts.

Par ailleurs, on distingue deux catégories de langages : les langages traditionnels et les langages web.

5.2 Les langages traditionnels

Dans cette catégorie nous avons les cinq principaux langages suivants :

5.2.1 Knowledge interchange format (KIF)¹

C'est un format d'échange de connaissances basé sur la logique de premier ordre. Il a été développé en 1992 pour résoudre les problèmes d'hétérogénéité entre les langages de représentation de connaissances d'une part et d'autre part, pour faciliter l'échange de connaissances entre les différents systèmes d'informations.

Bien que KIF offre des primitives pour représenter des objets, des relations et des fonctions, la construction d'ontologies avec ce langage est une tâche pénible. Ceci a mené à la création de OKBC et de « Frame Ontology » qui sont deux couches au dessus de KIF facilitant le développement d'ontologies.

5.2.2 Ontolingua²

C'est un langage d'ontologies créé par le laboratoire des systèmes de connaissances de l'université de Stanford. Il est construit au dessus de KIF et de la « Frame Ontology ». Ontolingua est basé sur la logique du premier ordre et le paradigme des cadres. Il est particulièrement utilisé par le serveur d'ontologies « Ontolingua Server » de l'université de Stanford.

¹ <http://logic.stanford.edu/kif/dpans.html>

² <http://www.ksl.stanford.edu/software/>

5.2.3 Loom

Il a été développé, entre 1986 et 1995, par des chercheurs du *Information Sciences Institute of Southern California University*. Avant d'être utilisé pour la construction d'ontologies, Loom a été créé initialement pour la construction de systèmes experts. Il est sous-jacent à la logique de description.

Par ailleurs, Loom est composé de deux sous-langages : le sous-langage de description et le sous-langage d'assertion. Le premier permet de décrire les modèles de domaine en utilisant les objets et les relations (TBox), alors que le deuxième est utilisé pour affirmer des faits concernant les instances (ABox).

5.2.4 OCML (Operational Conceptual Modeling Language)

Il a été développé en 1993 par le *Knowledge Media Institute* en Grande Bretagne dans le cadre du projet VITAL. OCML est basé sur le paradigme de cadres et il est compatible avec *Ontolingua*. Il fournit des primitives pour définir des classes, des fonctions, des axiomes et des instances.

5.2.5 FLOGIC (Frame Logic)

Il a été créé en 1995 par le département des sciences de l'informatique de l'université de New York. Il a été conçu comme une approche orientée objet de la logique de premier ordre. À l'origine, FLogic a été utilisé pour les bases de données déductives et les bases de données orientées objet. Par la suite, il a été adapté aux ontologies.

5.3 Les langages Web

5.3.1 Resource Description Framework (RDF)

À l'origine, le World Wide Web a été conçu pour partager des informations entre les humains. Ces informations sont présentées sous forme de pages « html » affichables par des navigateurs Web. Quelques années plus tard, cette conception s'est avérée très limitée. Cette limitation est due, d'une part au nombre important d'informations publiées sur le Web et d'autre part à leur hétérogénéité. En effet, le nombre colossal d'informations rend leur traitement manuel impossible; et l'hétérogénéité rend le traitement automatique très difficile.

Pour faire face à cette situation, des méta-données sont utilisées pour décrire les données et en faciliter le traitement. C'est dans ce contexte que le langage *Ressource Description Framework* est arrivé en 1999.

RDF est un modèle conceptuel permettant de décrire des « choses » selon un mécanisme simple et sans ambiguïté. Il est indépendant de toute syntaxe particulière. Cependant, l'organisation W3C a choisi la syntaxe XML pour utiliser RDF au sein du web sémantique, ce qui est appelé par certains auteurs RDF / XML.

5.3.1.1 Objectifs de RDF

L'organisation W3C a adopté le RDF dans les perspectives suivantes :

- a. définir un mécanisme pour décrire les ressources sur le Web en utilisant les métadonnées;
- b. faciliter le traitement automatique de l'information du Web par des agents logiciels;
- c. fournir l'interopérabilité entre les applications Web.

5.3.1.2 Modèle et syntaxe RDF

L'entité de base du modèle RDF est le triplet (sujet, prédicat, objet) représentant une **déclaration** « *statement* » sur une ressource :

- **Sujet** est la ressource que l'on veut décrire. On entend par ressource, toute chose exprimée avec RDF. Une ressource peut être une partie d'une page web, une page entière ou une collection de pages. Les ressources sont référencées par une URI (Uniform Resource Indicator) obligatoire et un identificateur de signet optionnel.
- **Prédicat** est une propriété de la ressource. Une propriété est un aspect, une caractéristique, un attribut, ou une relation spécifique utilisée pour décrire une ressource. Chaque propriété a une signification spécifique, des valeurs autorisées, des types de ressources qu'elle peut décrire et des relations avec d'autres propriétés.
- **Objet** est la valeur pour la propriété. Il peut être une autre ressource spécifiée par une URI, un littéral (une simple chaîne de caractères) ou un autre type de données primitif défini par XML.

Par exemple soit la phrase : « Ora Lassila est le créateur de la page <http://www.w3c.org/Home/Lassila> »¹.

Selon le modèle RDF, cette phrase est une déclaration constituée des éléments suivants :

- a. Sujet (ressource) : <http://www.w3c.org/Home/Lassila>;
- b. Prédicat (propriété) : créateur;
- c. Objet (valeur) : Ora Lassila.

On peut représenter cette déclaration par un graphe étiqueté orienté comme le montre la figure 5.1.

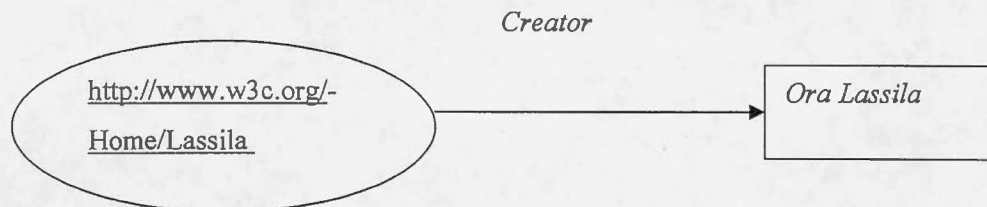


Figure 5.1 : Exemple d'un graphe étiqueté orienté

¹ <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>

Le document rdf/XML correspondant à cette déclaration est le suivant :

```
<rdf:RDF
xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
xmlns:dc="http://purl.org/metadata/dublin_core#">
<rdf:Description rdf:bout="http://www.w3.org/Home/Lassila">
<dc:Creator>Ora Lassila</dc:Creator>
</rdf:Description>
</rdf:RDF>
```

Figure 5.2: Déclaration d'un triplet RDF

Nous remarquons que tout document rdf commence par une balise ouvrante `< rdf:RDF>` et se termine par une balise fermante `< /rdf:RDF>`.

Les deux premières lignes déclarent les espaces de nom qui seront utilisés dans ce document *RDF* (obligatoire) et *Dublin Core* (optionel).

L'attribut « *about* » de l'élément « *Description* » permet de préciser la ressource qu'on veut décrire, il s'agit de la page web « <http://www.w3.org/Home/Lassila> ».

Enfin, l'élément *Creator* permet de spécifier le nom du créateur de la ressource concernée.

Par ailleurs, pour une machine, l'élément `<Creator>` comme tout autre élément défini par l'utilisateur n'a aucun sens particulier. Afin de donner du sens à ces éléments, on a étendu le langage RDF à RDF Schema (RDFS).

En effet, RDFS a un système de classes organisées en hiérarchie dont les ressources et les propriétés seront des instances. Ce système permet à l'aide d'un ensemble de constructeurs de :

- donner du sens aux propriétés associés aux ressources;
- formuler des contraintes sur les valeurs associées à une propriété;
- préciser le type des propriétés, leur domaine .etc.

5.3.1.3 Les principaux constructeurs de RDF(S)

- **rdfs:class** pour déclarer une classe.
- **rdfs:subClassOf** pour définir des hiérarchie de classes.
- **rdfs:domain, rdfs:range** : pour limiter les valeurs possibles des propriétés.
- **rdf:type** : pour déclarer une ressource comme instance d'une classe spécifique.
- **rdf:property** : pour déclarer une propriété.
- **rdf:Datatype** : pour spécifier le type de donné.

5.3.2 DAML+OIL¹

C'est un langage de représentation des ontologies, il a vu le jour après la fusion des deux langages DAML (DARPA Agent Markup Language) et le langage OIL (Ontology Inference Layer).

DAML est apparu en 2000 suite à un programme du Ministère de la Défense américaine. Son objectif consistait à résoudre les problèmes d'interopérabilité sur le Web et à faciliter le Web sémantique.

OIL est le premier langage d'ontologies, il est basé sur XML et RDF(S) et il implémente les deux paradigmes de représentation des connaissances, *la logique de description et les cadres « frames »*.

À l'instar de RDF(S), DAML+OIL permet d'exprimer des classes, des propriétés, des instances ainsi que des relations entre les classes. De plus, DAML+OIL étend RDF(S) par des nouveaux constructeurs pour exprimer des relations plus complexes. DAML+OIL constitue la base du langage OWL, le nouveau standard du W3C que nous avons utilisé pour développer l'ontologie OCJM et qui est présenté à la section suivante.

¹ <http://www.w3.org/TR/daml+oil-reference>

5.3.3 Le Web Ontology Language²

Le Web Ontology Language (OWL) est un langage XML permettant d'écrire, de publier et de partager des ontologies sur le Web. OWL est basé sur RDF(S) et inspiré du langage DAML-OIL. Il est devenu un standard du W3C en février 2004. Il est conçu pour être compatible avec l'architecture du Web en général et du Web sémantique en particulier. Si RDF(S) permet la description des classes et des propriétés, OWL apporte en plus la possibilité de comparer des classes et des propriétés. En fournissant un grand nombre de constructeurs, OWL permet d'exprimer des relations plus complexes entre les classes et des contraintes plus précises sur les classes et sur les propriétés (W3C, 2004).

Ainsi, les ontologies exprimées en OWL possèdent les avantages suivants (W3C, 2004) :

- la capacité d'être distribuées au travers de nombreux systèmes;
- la mise à l'échelle pour les besoins du Web;
- la compatibilité avec les standards du Web;
- l'ouverture et l'extensibilité.

5.3.3.1 Les trois sous-langages de OWL

Le langage OWL fournit trois sous-langages d'expressivité croissante, destinés à des communautés différentes de développeurs.

a. *OWL Lite* est le sous-langage de OWL le plus simple. Il est destiné aux utilisateurs ayant principalement besoin d'une hiérarchie de classification et de contraintes simples. Par exemple, les contraintes de cardinalité sont limitées à 0 ou à 1 seulement.

b. *OWL DL* est plus complexe que OWL Lite. Il est destiné aux utilisateurs qui demandent une expressivité plus importante. OWL DL comprend toutes les structures du langage OWL mais utilisables avec certaines restrictions. Par exemple une instance d'une classe ne peut être elle-même une classe (la séparation des types).

OWL DL se nomme ainsi en raison de ses liens avec la logique de description.

² W3C a choisit l'acronyme OWL au lieu de WOL

c. *OWL Full* est le sous-langage de OWL le plus complexe. Il est destiné aux utilisateurs qui veulent une expressivité maximale. Dans OWL Full les restrictions de *OWL DL* sont levées.

Le groupe *WebOnto* (*Web-Ontology Working Group*) de W3C souligne qu'il est peu probable qu'un système de raisonnement puisse mettre en oeuvre toutes les caractéristiques de OWL Full.

Chacun de ces sous-langages représente une extension par rapport à son prédécesseur plus simple. Ainsi, les affirmations suivantes sont vraies et leurs symétriques ne le sont pas.

- Toute ontologie OWL Lite légale (correcte) est une ontologie OWL DL légale.
- Toute ontologie OWL DL légale est une ontologie OWL Full légale.
- Toute inférence (conclusion) OWL Lite valide est une inférence OWL DL valide.
- Toute inférence OWL DL valide est une inférence OWL Full valide.

5.3.3.2 La structure d'une Ontologie OWL

Une ontologie OWL est présentée dans un « document OWL » sous forme de fichier texte. L'extension dudit fichier selon les recommandations du groupe *WebOnto* de W3C, est *rdf* ou *owl*.

a. Les espaces de nom

Une ontologie OWL commence typiquement par des déclarations d'espaces de nom (ou espace de nommage). Ces déclarations permettent d'indiquer la provenance des termes utilisés afin de les interpréter sans ambiguïté. Elles sont contenues dans une balise **rdf:RDF**.

Voici un exemple de déclarations d'espace de nom pour une ontologie de *wine* :


```
<rdf:RDF
```

- 1) xmlns =" <http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#> "
- 2) xmlns:wine =" <http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#> "
- 3) xml:base =" <http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#> "
- 4) xmlns:food =" <http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#> "
- 5) xmlns:owl =" <http://www.w3.org/2002/07/owl#> "
- 6) xmlns:rdf =" <http://www.w3.org/1999/02/22-rdf-syntax-ns#> "
- 7) xmlns:rdfs =" <http://www.w3.org/2000/01/rdf-schema#> "
- 8) xmlns:xsd =" <http://www.w3.org/2001/XMLSchema#> ">

Figure 5.3 : Déclaration d'espace de nom en OWL (W3C, 2004)

La première ligne déclare l'ontologie courante comme espace de nom **implicite** en déclarant que les noms qualifiés sans préfixe se rapportent à l'ontologie courante.

La deuxième ligne identifie l'espace de nom de l'ontologie courante au préfixe *wine* :

La troisième ligne identifie l'adresse URI de base de ce document.

La quatrième ligne identifie l'espace de nom de l'ontologie de soutien *food* au préfixe *food* (dans l'ontologie *wine* on utilisera des concepts et des termes définis dans l'ontologie *food*).

La cinquième ligne dit que les éléments préfixés par **owl** : devraient se comprendre comme se référant aux choses issues de l'espace de nom appelé « <http://www.w3.org/2002/07/owl#> » C'est une déclaration OWL conventionnelle qui introduit le vocabulaire OWL.

Les trois dernières lignes (6,7 et 8) déclarent les espaces de nom de RDF, de RDFS et de type de données du schéma XML.

L'écriture des adresses peut être simplifiée en utilisant les déclarations de type de document (DOCTYPE). Ainsi, l'exemple précédent peut être simplifié comme suit :

```
<!DOCTYPE rdf:RDF [  
  <! ENTITY wine "http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine#" >  
  <!ENTITY food "http://www.w3.org/TR/2004/REC-owl-guide-20040210/food#" > ]>  
<rdf:RDF  
  xmlns    ="&wine;"  
  xmlns:wine ="&wine;"  
  xml:base ="&wine;"  
  xmlns:food ="&food;"  
  xmlns:owl ="http://www.w3.org/2002/07/owl#"  
  xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"  
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"  
  xmlns:xsd ="http://www.w3.org/2001/XMLSchema#">
```

Figure 5.4 : Exemple d'un document *DOCTYPE*

b. Les en-têtes

Après la déclaration des espaces de nom, on écrit un en-tête contenant les métadonnées sur l'ontologie. Ces métadonnées contiennent, entre autres, des commentaires, la version de l'ontologie et l'importation (l'inclusion) éventuelle d'autres ontologies. C'est la balise **owl : *Ontology*** qui permet d'écrire l'en-tête de l'ontologie comme le montre l'exemple suivant (fig. 5.5).

```
<owl:Ontology rdf:about=""/>
<rdfs:comment>Un exemple d'ontologie OWL</rdfs:comment>
<owl:priorVersion rdf:resource="http://www.w3.org/TR/2003/PR-owl-guide-
20031215/wine"/>
<owl:imports rdf:resource="http://www.w3.org/TR/2004/REC-owl-
guide-20040210/food"/>
<rdfs:label>Ontologie sur le vin</rdfs:label>
</owl:Ontology>
```

Figure 5.5 : Exemple d'en-tête d'ontologie OWL

L'attribut "**rdf:about**" fournit un nom ou une référence pour l'ontologie. Lorsque sa valeur est "", comme c'est le cas habituellement, le nom de l'ontologie vaut l'adresse URI de base de l'élément owl : ontology (sect. 5.3.3.2 a). C'est en générale l'adresse URI du document contenant l'ontologie. Une exception serait un contexte utilisant l'attribut "**xml:base**", qui peut assigner à un élément une adresse URI de base autre que celle du document courant.

Il est important de distinguer l'inclusion des ontologies de la déclaration des espaces de nom. Cette dernière permet d'utiliser des entités définies dans d'autres ontologies, alors que l'inclusion permet d'utiliser ces entités et d'inférer à partir d'elles. Par ailleurs, l'inclusion est transitive (l'inclusion de l'ontologie O1 inclut toutes les ontologies incluses dans O1).

c. Les éléments de bases de OWL

Les éléments de base d'une ontologie OWL sont principalement les classes, les instances (individus) et les propriétés. Une classe est un ensemble d'individus qui ont des caractéristiques similaires. Les individus représentent les instances de la classe. Les caractéristiques et des éventuelles relations entre individus représentent les propriétés.

Il est à noter que *OWL Full* est le seul sous-langage de OWL permettant qu'une classe puisse être l'instance d'une autre classe, contrairement à *OWL Lite* et *OWL DL*.

Les principaux constructeurs de OWL

En plus des constructeurs de RDF(S), OWL dispose d'un grand nombre de constructeurs qui permettent d'exprimer des relations plus complexes entre les classes et des contraintes plus précises sur les classes et sur leurs propriétés.

Dans ce paragraphe, nous présentons les principaux constructeurs fournis par OWL regroupés en trois catégories : les classes, les propriétés et les individus (instances).

1. Classes

Les principaux constructeurs de cette catégorie sont :

- **owl:Class** : Dans une ontologie OWL, les concepts élémentaires d'un domaine sont représentés par des classes. On déclare une classe à l'aide de la balise **<owl:Class>** et en lui attribuant un identificateur unique qui est la valeur de l'attribut

rdf:ID= « nom de la classe ».

Ainsi une classe « **Personne** » se déclare de la façon suivante : **<owl:Class rdf:ID="Personne"/>**

- **rdfs:subClassOf** : les classes peuvent être organisées en une hiérarchie de spécialisation à l'aide du constructeur *subClassOf*. Ce constructeur relie une classe spécifique à une classe plus générale. Si X est une sous-classe de Y, alors chaque instance de X est également une instance de Y. La relation *subClassOf* est transitive.

Chaque classe définie par le développeur est implicitement une sous-classe d'une classe intégrée nommée *Thing*. Par ailleurs, la classe nommée *Nothing* est une sous-classe de toutes les classes OWL et elle ne peut avoir aucune instance.

- **owl:equivalentClass** : on peut déclarer deux classes comme étant équivalentes. Les classes équivalentes ont les mêmes instances. Par exemple, on peut déclarer la classe « Voiture » comme équivalente à la classe « Automobile ».
- **owl:oneOf (DL, Full)**: on peut décrire une classe par l'énumération des individus qui la composent. Les membres de la classe sont exactement l'ensemble des individus énumérés.
- **owl:disjointWith (DL, Full)** : on peut déclarer les classes comme étant disjointes les unes des autres.
- **owl:unionOf, owl:complementOf, owl:intersectionOf (DL, Full)** : combinaison booléennes de classes.

2. Propriétés

OWL dispose de deux catégories distinctes de propriétés :

- les propriétés d'objets permettant de relier des instances (individus) à d'autres instances
- les propriétés de type de données permettant d'assigner des valeurs de données à des instances.

Une propriété d'objet est une instance de la classe owl:ObjectProperty, alors qu'une propriété de type de données est une instance de la classe owl:DatatypeProperty. Ces deux classes sont elles mêmes sous-classes de la classe RDF rdf : Property.

Les principaux constructeurs de la catégorie propriétés sont :

- **subPropertyOf** : On peut créer des hiérarchies de propriétés en déclarant une propriété comme sous-propriété d'une ou de plusieurs autres propriétés.
- **rdfs:domain** : le domaine d'une propriété limite les individus auxquels la propriété s'applique.
- **rdfs:range** : c'est l'image d'une propriété, elle limite les individus ou les valeurs de données qui fournissent la valeur d'une propriété.
- **owl:equivalentProperty** : ce constructeur sert à déclarer deux propriétés comme étant équivalentes.
- **owl:inverseOf** : on l'utilise pour déclarer qu'une propriété peut être déclarée comme étant la symétrique d'une autre.

- **owl:TransitiveProperty** : on peut déclarer une propriété comme étant transitive.
- **owl:SymetricProperty** : on peut déclarer une propriété comme étant symétrique.
- **owl:FunctionalProperty** : une propriété de la classe functionalProperty n'a pas plus d'une seule valeur pour chaque individu (un individu peut n'avoir aucune valeur)
- **owl:minCardinality** : pour préciser le nombre minimal de valeurs associées à une propriété.
- **owl:maxCardinality** : pour préciser le nombre maximal de valeurs associées à une propriété.
- **owl:cardinality** : pour préciser le nombre exact de valeurs associées à une propriété.
- **owl:hasValue (DL, Full)** : on peut obliger une propriété à avoir un certain individu comme valeur.

3. Individus

Les constructeurs de cette catégorie permettent de déclarer et de définir des instances (individus) de classes et les relations qui les relient.

- **owl:sameAs** : on peut déclarer deux individus comme étant les mêmes. Ce constructeur s'utilise pour créer un certain nombre de noms différents qui se rapportent au même individu.
- **owl:differentFrom** : on peut déclarer un individu comme étant différent des autres individus.
- **owl:allDifferent** : on peut déclarer un certain nombre d'individus comme étant mutuellement distincts.

5.3.4 Différences entre DAML+OIL et OWL

Les principales différences entre les langages OWL et DAML+OIL sont :

1. Les restrictions qualifiées ont été supprimées du langage, en effet OWL n'inclut pas les propriétés : `daml:cardinalityQ`, `daml:hasClassQ`, `daml:maxCardinalityQ` et `daml:minCardinalityQ`.
2. Plusieurs primitives (propriétés et classes) ont été renommées comme l'indique le tableau 5.1.
3. À l'inverse de DAML+OIL, OWL permet de définir des propriétés symétriques grâce à la classe `owl:SymmetricProperty` qui a été ajoutée.
4. OWL ne renomme pas les classes et les propriétés du langage RDF et du schéma RDF utilisées. Par exemple, `rdf:subClassOf` et `rdf:type`
5. Les propriétés `owl:AllDifferent` et `owl:distinctMembers` ont été ajoutées afin de répondre à la présomption des noms uniques.

Au cours de ce chapitre, nous avons présenté et expliqué les quatre principaux paradigmes de représentation des connaissances sur lesquels se basent les langages d'ontologies. Ces paradigmes sont la logique du premier ordre, la logique de description, les *frames* et les graphes conceptuels. Nous avons aussi présenté les principaux langages regroupés en deux catégories : les langages classiques et les langages Web dont nous avons détaillé la présentation. Nous avons particulièrement détaillé la présentation du OWL, le standard le plus récent de W3C que nous avons utilisé pour exprimer l'ontologie OCJM. Cette dernière sera développée au chapitre suivant.

Tableau 5.1 : Primitives DAML+OIL renommées en OWL (W3C, 2004)

DAML+OIL	OWL
daml:differentIndividualFrom	owl:differentFrom
daml:equivalentTo	owl:sameAs
daml:sameClassAs	owl:equivalentClass
daml:samePropertyAs	owl:equivalentProperty
daml:hasClass	owl:someValuesFrom
daml:toClass	owl:allValuesFrom
daml:UnambiguousProperty	owl:InverseFunctionalProperty
daml:UniqueProperty	owl:FunctionalProperty

CHAPITRE VI

L'ONTOLOGIE « OCJM »

Dans ce chapitre, nous présenterons en détail et à l'aide d'exemples, la conception et la formalisation de l'ontologie du centre de jeunesse de la Montérégie (OCJM).

6.1 Conceptualisation de l'ontologie OCJM

En suivant la méthodologie proposée par Natalya et McGuinness (chap. 4), nous avons commencé par la définition du domaine et la portée de l'ontologie en répondant à des questions dites de « compétences » telles que :

- À quels types de questions l'ontologie devra-t-elle fournir des réponses?
- Quels sont les utilisateurs potentiels de notre ontologie?

En répondant à ces questions, nous avons constaté que les connaissances et les informations représentées dans l'ontologie ne devraient pas être très spécialisées mais d'ordre général. En d'autres termes, ces connaissances sont habituellement utilisées au CJM et par conséquent elles devraient être connues et comprises par tous les employés. Nous avons donc constaté que l'ontologie devrait avoir au moins les six parties ou sections suivantes (fig. 6.1): **CJM, Personne, Médecine familiale, Sciences infirmières, Psychologie et Éducation.**

Cependant, étant donné la grande étendue de l'ontologie OCJM et la disponibilité des employés et des documents, nous avons préféré nous focaliser seulement sur les trois premières sections. Par ailleurs, les trois autres sections pourraient être ajoutées à l'ontologie sans affecter aucunement le système comme ce sera expliqué au chapitre suivant.

CJM	Médecine familiale	Personne
Psychologie	Sc. infirmières	Éducation

Figure 6.1: Les sections de l'ontologie OCJM

La section qui suit présente les trois sections de l'ontologie que nous avons retenues en plus de la section « génogramme » (fig. 6.2).

1. CJM

Cette partie décrit la structure organisationnelle du CJM, ses départements et les services qui y sont offerts.

2. Personne

Dans cette partie nous décrivons les deux principales catégories de personnes soient les patients et les employés du CJM. On y trouve des renseignements personnels comme le nom et le prénom d'une personne, le poste occupé par un employé ainsi que son expertise etc.

3. Médecine familiale

Cette partie est consacrée aux connaissances du domaine de la médecine familiale. Elle contient les principaux termes et concepts représentant les différentes branches de la médecine familiale. Ces termes et concepts sont organisés en une hiérarchie de classes.

Nous avons construit cette partie en nous basant sur le document de Bernstein et al. (2001).

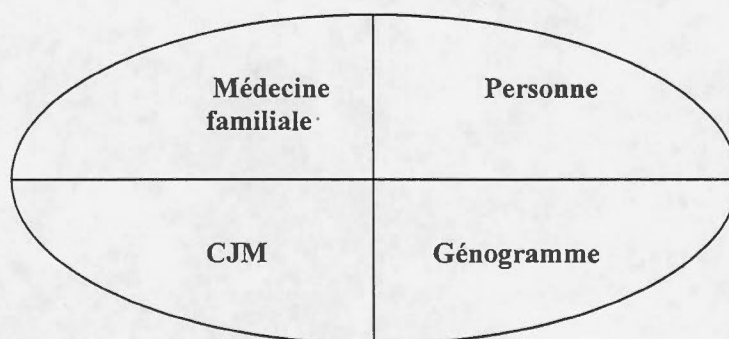


Figure 6.2 : Les ~~part~~ sections de l'ontologie OCJM retenues

4. Génogramme

Bien que le génogramme soit un outil utilisé par les médecins de famille (chap. 1), nous avons préféré lui consacrer une partie entière de l'ontologie à cause de son importance et de sa large utilisation. De plus, il est constitué de plusieurs symboles dont la connaissance est nécessaire pour comprendre les dossiers médicaux des clients.

Par la suite, nous avons établi la liste des principaux termes utilisés au sein du CJM (tabl. 6.1; appendice A). Par la suite, ces termes ont été classés selon les quatre parties de l'ontologie décrites ci-dessus.

L'étape suivante est de définir les classes et la hiérarchie des classes. Pour ce faire, il existe trois approches (fig. 6.3; appendices B et C) :

- l'approche descendante : on commence par les concepts les plus généraux et on descend vers les concepts les plus spécifiques;
- l'approche ascendante : on commence par la définition des classes les plus spécifiques, ensuite les classes les plus générales ou abstraites;
- l'approche mixte : on combine les deux approches ci-dessus.

C'est cette troisième approche que nous avons utilisée car au début de projet nous n'avions pas une vue systématique du CJM et de ses activités.

À l'étape suivante nous avons défini les propriétés et les attributs intrinsèques et extrinsèques des classes ainsi que les relations entre les classes. Ensuite, nous avons défini les facettes des attributs (cardinalité, domaine, rang... etc.) et nous avons créé quelques instances de classes afin de tester notre prototype. Finalement, nous avons fait valider l'ontologie OCJM par notre directeur et notre codirecteur.

Tableau 6.1 : Exemple de termes utilisés au CJM

Enfant	Antécédents personnels,	Relation d'union
Jeune	Médicaux et psychiatrique	Relation affective
Père	Histoire sociale	Relation de descendance
Mère	Occupation, travail à risque	Relation d'ascendance
Parent	Famille	Relation de jumeau
Grand parant	Sport et loisirs	Adoption
Grand père maternel	Habitude de vie	Les symboles
Grande mère maternelle	Alimentation	Situation familiale
Grand père paternel	Tabac	Patterns
Grand mère paternel	R-OH, caféine, drogue	CLSC
Conjoint	Voyage	Cliniques
Conjointe	Médication	Médecin
Frère	Risque élevé	Thérapeute
Sœur	Génogramme	Psychologue
Fille	Génogramme informatisé	Famille complexe
Fils	Chirurgie	Famille recomposée
Neveu	Allergie	Parents biologiques
Nièce	Vaccination	Parents légaux
Petit-fils	Les professionnels de la santé	Couple homosexuel
Petite-fille	Travailleurs sociaux	Enfant adopté
Petit-neveu	Médecin de famille	Vue systémique
Petite-nièce	Infirmier	L'histoire médicale de la famille
Amis	Agent d'administration	Maladies héréditaires
Décédé	Thérapie familiale	Mal. psychiatrique
Année de naissance	Recherche clinique	Allergie
Lien de parenté	Héritage	Mal. cardiaque
Arbre généalogique	Théorie des systèmes familiaux	Mal. Neurologique
Antécédents familiaux	Arbre généalogique	Patient
Glaucome	Individu	Client
Tuberculose	Masculin	Consultation
Diabète	Féminin	Centre de jeunesse
Cancer	Structure familiale	Service
Goutte	Avortement provoqué	Contexte social
hyperlipidémie	Avortement spontané	Contexte familial
HTA	Mort-né	Événement
État émotionnel	Grossesse	Dossier médical
État physique	Relation	Modèles familiaux
Centre de santé et	Protection des jeunes	Condition de vie
Services sociaux	Prévention	Trouble psychologique
Hôpital	Signalement	Garde partagée
Hébergement	Urgence sociale	etc.
Réadaptation	Orientation des jeunes	

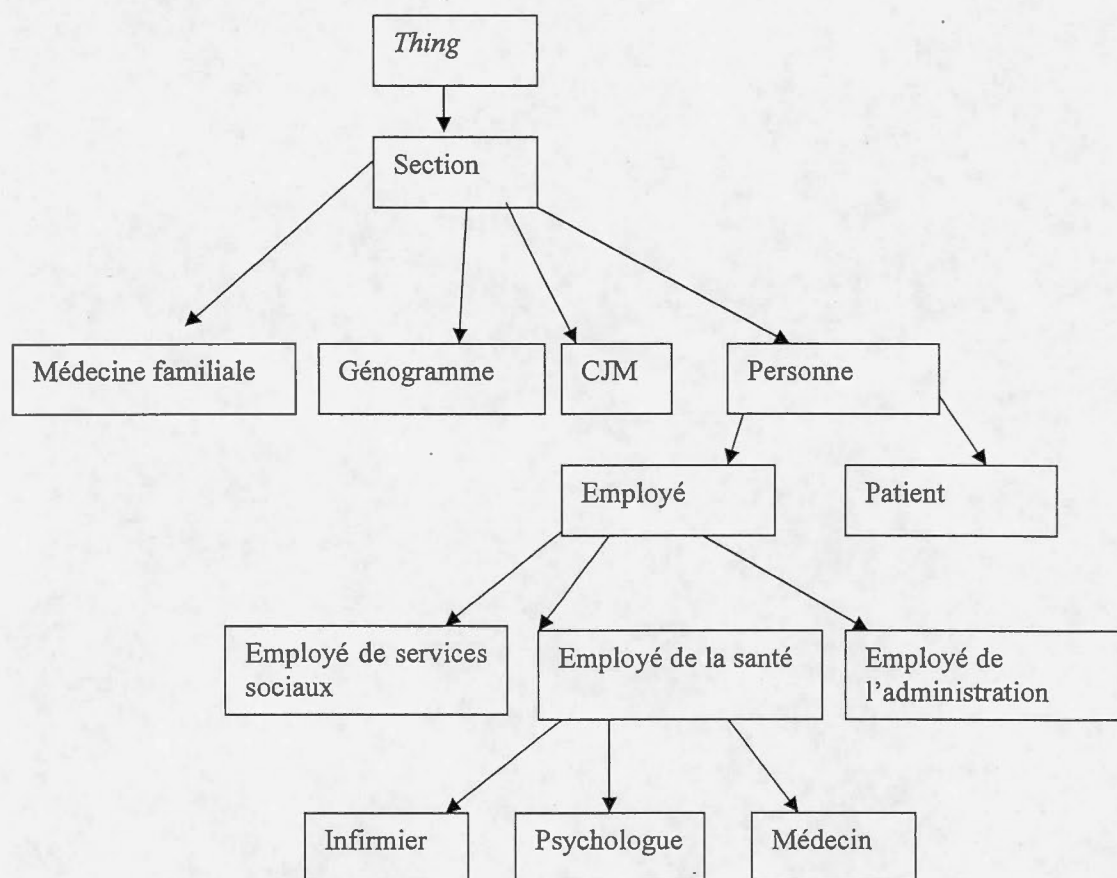


Figure 6.3 : Exemple d'hérarchie de classes

6.2 La formalisation de l'ontologie

Afin d'intégrer et de déployer l'ontologie dans le système de gestion des connaissances, nous l'avons rendue formelle en l'exprimant avec le langage OWL DL (appendice D).

6.2.1 L'en-tête et les espaces de nom

Nous avons commencé par la déclaration des espaces de nommage qui sont utilisés dans l'ontologie, ainsi que le titre et la version de l'ontologie.

```
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns="http://localhost:8080/monApp/CJM.owl#"
xml:base="http://localhost:8080/monApp/CJM.owl#">
```

Figure 6.4 : Les espaces de nom de

```
<owl:Ontology rdf:about="">
<owl:versionInfo>$Id: OCJM.owl,v 1.0 2006/03/15 Exp $</versionInfo>
<rdfs:comment >
  Ontologie du centre de jeunesse de la Montérégie  OCJM
</rdfs:comment>
</owl:Ontology
```

Figure 6.5 : L'en-tête de l'ontologie OCJM

6.2.2 Définition des classes

Après l'en-tête et les espaces de nommage, nous avons défini la hiérarchie des classes (appendices B et C). Par exemple, la classe *Section* est la classe principale de l'ontologie OCJM et elle est sous-classe de la classe *Thing*. En même temps, *Section* est la classe mère (super-classe) des classes *Personne* et *CJM*

```
<owl:Class rdf:ID="Section">
<rdfs:comment >
La classe principale de l'ontologie OCJM
</rdfs:comment>
</owl:Class>
<owl:Class rdf:ID="Personne">
<rdfs:subClassOf>
<owl:Class rdf:ID=" Section "/>
</rdfs: subClassOf>
</owl: Class>
<owl:Class rdf:ID="CJM">
<rdfs:subClassOf>
<owl:Class rdf:ID=" Section"/>
</rdfs:subClassOf>
</owl:Class>
```

Figure 6.6 : Exemple de déclaration de classes

6.2.3 Définition de propriétés

Après la définition des classes, nous avons défini les propriétés d'objets et les propriétés de types de données. Par exemple, la classe **Personne** contient les propriétés **nom**, **prénom** **âge** etc. (appendice C).

6.2.4 Définition des instances

En général, la définition d'une instance consiste premièrement à déclarer l'appartenance d'un individu à une classe et deuxièmement à déclarer les valeurs de ses propriétés.

OWL permet également de définir des instances à l'aide des propriétés telles que : *sameAs* et *differentFrom*. Dans notre cas nous avons déclaré quelques instances de la classe **Personne** (appendice C).

Une fois l'ontologie OCJM écrite en langage OWL, nous l'avons intégrée dans le SGC comme nous le verrons au chapitre suivant, consacré à l'implémentation du prototype.

CHAPITRE VII

IMPLÉMENTATION DU PROTOTYPE

Avant l'implémentation de notre prototype, nous avons conçu son architecture, défini ses modules, choisi les technologies adéquates et les outils efficaces à utiliser. La plupart de ces technologies et outils ont été appris lors de ce projet de recherche. L'incompatibilité de certains outils et technologies ainsi que des problèmes de versions figurent parmi les difficultés techniques que nous avons surmontées.

7.1 Modules du prototype

Le prototype du système de gestion des connaissances du CJM est constitué de quatre principaux modules (composantes) suivants (fig. 7.1) :

- a. L'ontologie OCJM écrite en langage OWL.
- b. L'application pour charger l'ontologie OCJM dans la base de connaissances en utilisant un programme java et l'API Jena (sect. 7.3.5).
- c. La base de connaissances constituée de l'ontologie OCJM et ses instances et de la BD *utilisateurs*. Elle utilise le serveur de base de données MySQL.
- d. L'application Web pour naviguer dans la base de connaissances et explorer l'ontologie OCJM. Cette application est constituée de modules JavaBeans et des pages JSP, en plus d'un annuaire électronique constitué de pages Web annotées comme ce sera expliqué dans le guide d'utilisation plus loin dans ce chapitre.

Les sections suivantes expliquent en détail les composants et les modules du prototype.

7.2 L'architecture technique

Vu la complexité de notre application, nous avons opté pour une architecture à trois niveaux ou architecture 3-tiers (fig. 7.2). Cette architecture caractérise les systèmes d'information client/serveurs où il existe un niveau intermédiaire entre le client et le serveur :

- a. **Le client** est un ordinateur qui demande une ressource en utilisant une interface utilisateur (dans notre cas un navigateur web) chargée de la présentation ;
- b. **Le serveur d'application** ou *middleware* (le serveur Tomcat dans notre application) est chargé de fournir la ressource demandée en faisant appel à un autre serveur;
- c. **Le serveur de données** (MySQL dans notre application) fournit au serveur d'application les données dont il a besoin pour répondre à la demande du client.

Cette architecture est choisie pour les avantages suivants :

- o Une grande flexibilité et souplesse pour la maintenance et l'évolution de l'application;
- o Une sécurité accrue qui peut être définie indépendamment à chaque niveau et pour chaque service;
- o De meilleures performances, car les tâches et les services sont partagés entre les différents serveurs.

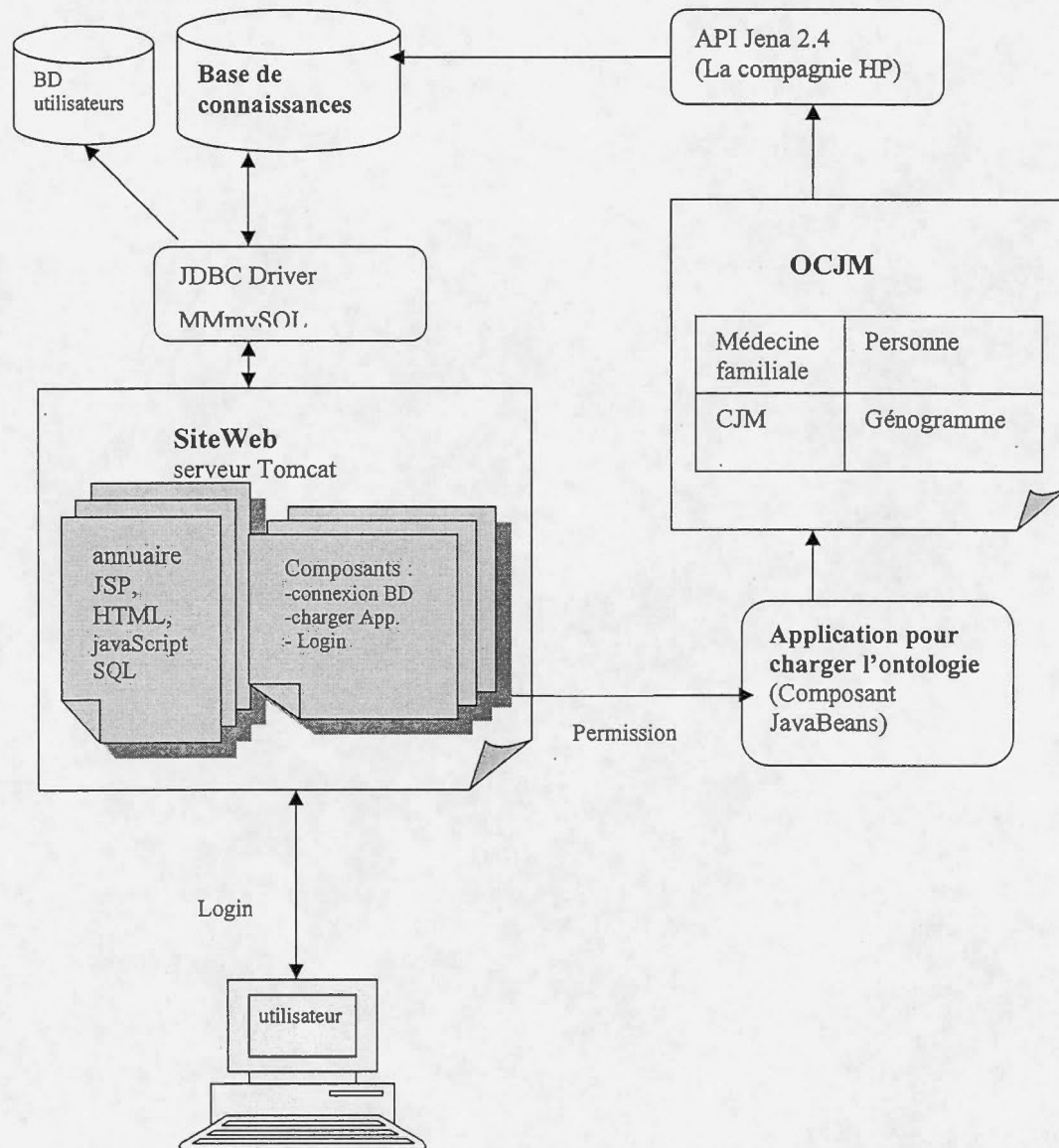


Figure 7.1 : Les composantes du SGC

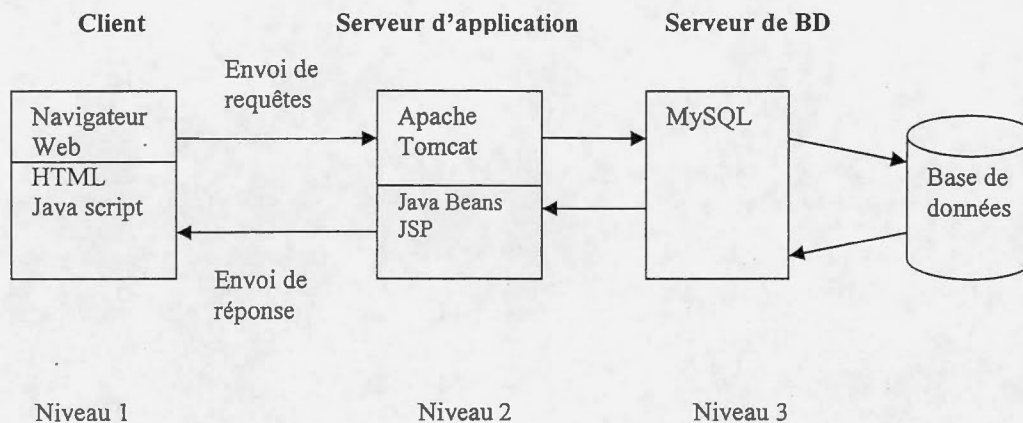


Figure 7.2 : L'architecture technique du SGC

7.3 Les technologies et les langages utilisés

7.3.1 Java

Java est à la fois un langage de programmation orienté objet et une plate-forme d'exécution JRE (*Java Runtime Environment* - Environnement d'exécution Java). Le langage Java a la particularité principale d'être portable sur plusieurs systèmes d'exploitation tels que Windows ou Linux. C'est la plate-forme qui garantit la portabilité des applications développées en Java. Java permet de développer des applications client/serveur autonomes, pouvant être exécutées sur tous les systèmes d'exploitation pour lesquels a été développée une plate-forme Java. Cette dernière est constituée d'une JVM (*Java Virtual Machine* - Machine Virtuelle Java) et d'un programme qui interprète le code Java et le convertit en code natif. JRE est aussi constitué d'une bibliothèque standard à partir de laquelle doivent être développés tous les programmes en Java. (Appendice E).

7.3.2 JavaBeans

Un JavaBean (ou simplement Bean) est un composant logiciel réutilisable, écrit en Java (appendice E). Il est comparable aux contrôles ActiveX de Microsoft. Les Beans sont destinés à être utilisés dans des programmes Java classiques, ou des application Web utilisant les servelets ou les JSP. Le développement des JavaBeans doit respecter certaines conventions sur le nommage des méthodes, la construction et le comportement. Le respect de ces conventions rendra possible l'utilisation et la réutilisation de JavaBeans dans des applications Java.

7.3.3 Java Server Pages (JSP)

C'est un langage de script puissant, exécuté du côté serveur et non du côté client. Les JSP permettent de développer des applications Web interactives dont le contenu est dynamique (Appendice F). Le contenu des pages Web JSP change selon certains paramètres tels que les informations stockées dans une base de données ou les préférences de l'utilisateur.

Les JSP sont intégrables dans une page Web HTML à l'aide de balises spécifiques. Ces balises permettent au serveur Web de savoir que le code compris à l'intérieur d'elles doit être interprété pour renvoyer le code HTML correspondant au navigateur du client.

Étant donnée que les JSP sont basées sur Java, elles possèdent toutes les caractéristiques faisant la force de Java : les JSP sont portables, sont orientées objet, sont sûres.etc.

7.3.4 JDBC

La technologie JDBC (*Java DataBase Connectivity*) est une API (*Application Programming Interface*) qui permet de se connecter à des bases de données, directement à partir des programmes Java. En plus d'être indépendant de la plate-forme sur laquelle elle s'exécute, l'API JDBC est également indépendante des SGBDs. En effet, elle a été développée de façon à permettre à un programme de se connecter à n'importe quelle base de données en utilisant la même syntaxe.

La version de pilote JDBC que nous avons utilisé est la 3.1 plutôt que la 5.0, car cette dernière n'est pas encore compatible avec l'API Jena.

7.3.5 API Jena

Jena est un *framework* Java qui aide à développer les applications du Web sémantique. Elle est développée au sein du Projet open source de *HP Labs Semantic Web Program*. Les premières versions de Jena (V1.X, 2000) fournissent des outils pour manipuler les données représentées en RDF, RDFS et DAML + OIL. À partir de sa deuxième version (V2.0, 2003), elle permet de manipuler des ontologies écrites en OWL. L'API Jena est utilisée dans 70% des applications du Web sémantique. La dernière version de Jena est la (V2.4, 2006), mais nous avons utilisé la version (V2.3, 2005) car elle est plus stable et parfaitement compatible avec MySQL 5.0 et OWL.

7.3.6 MySQL ¹

MySQL est un système de gestion des bases de données *open source*, développé sous double licence : licence GPL (*General Public License*) et licence commerciale. Ce serveur de base de données est très rapide, multi-thread, robuste et multi-utilisateurs. Il fonctionne sur beaucoup de plates-formes incluant Linux et Windows 95,98, NT, 2000 et XP.

Nous avons utilisé la version 5.0 offerte sous licence GPL.

7.3.7 Serveur Tomcat ²

Le serveur Tomcat est un serveur d'application *Open Source*, il fait partie du projet Jakarta, au sein de la fondation Apache. Tomcat implémente les spécifications des servlets et des JSP de *Sun Microsystems*. Comme Tomcat inclut un serveur HTTP interne, il est aussi considéré comme un serveur HTTP. Puisque Tomcat a été développé en langage Java, il peut donc s'exécuter via la JVM sur n'importe quel système d'exploitation. La version 5.5, que nous avons utilisée, implémente les servlets V2.4 et les JSP V2.0.

¹ <http://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html>

² <http://tomcat.apache.org/>

7.3.8 OWL

Le Web Ontology Language (OWL) est un langage XML permettant d'écrire, de publier et de partager des ontologies sur le Web. OWL est basé sur RDF(S) et inspiré du langage DAML+OIL. Il est devenu le langage standard de W3C en février 2004. Il est conçu pour être compatible avec l'architecture du Web en général et du Web sémantique en particulier (chap. 5).

7.3.9 Protégé 2000

Protégé 2000 est un système permettant la création d'ontologies. Il a été créé à l'Université de Stanford et est très populaire dans le domaine du Web Sémantique aussi qu'au niveau de la recherche académique. Il est développé en Java et offert sous licence GPL et à code source libre. Protégé peut lire et sauvegarder des ontologies dans la plupart des formats d'ontologies : RDF, RDF-S, et DAML-OIL et récemment OWL.

7.3.10 JCreator

JCreator est un puissant environnement de développement logiciel (*IDE : Integrated Development Environment*) pour java. À l'aide d'une interface facile et intuitive pour les habitués de Windows, JCreator permet au développeur de gérer les projets, visualiser les classes, déboguer du code Java.etc. Il est offert en deux versions : commerciale et gratuite. C'est cette dernière version que nous avons retenue.

7.4 Le déploiement du SGC

Le système est une application Web autonome « stand alone » qui peut être hébergée dans n'importe quel serveur muni d'une machine virtuelle Java, d'un serveur de base de données MySQL et d'un serveur d'application Web Tomcat de Apache. Pour utiliser le système, il suffit donc de taper l'adresse URI correspondante comme dans le cas de n'importe quel site Web.

Par ailleurs, il suffit de charger l'ontologie une seule fois et non pas chaque fois qu'on utilise l'application.

De plus, on pourra modifier l'ontologie en modifiant le fichier texte « OCJM.owl » correspondant qui n'est pas obligatoirement sauvegardé sur le même serveur que le reste de l'application (appendice D). Cette modification est faisable en utilisant n'importe quel éditeur

de texte. Après la modification il faudra recharger l'ontologie dans la BD en exécutant l'application « charger_ontologie » dont un lien hypertexte est disponible sur la page d'accueil des utilisateurs disposant d'une permission (fig. 7.13).

Le reste de ce chapitre constitue un guide d'utilisation du prototype de SGC. Il explique les différentes fonctionnalités offertes à l'aide des scénarios d'utilisation.

7.5 Guide d'utilisation du prototype

Une fois que l'utilisateur a accédé au site Web, il pourra naviguer selon le schéma simplifié de la figure 7.3. La page d'accueil offre trois possibilités : la description du prototype, le guide d'utilisation et le *login* qui permet de commencer l'exploration des connaissances stockées dans l'ontologie et consulter l'annuaire électronique (fig.7.4).

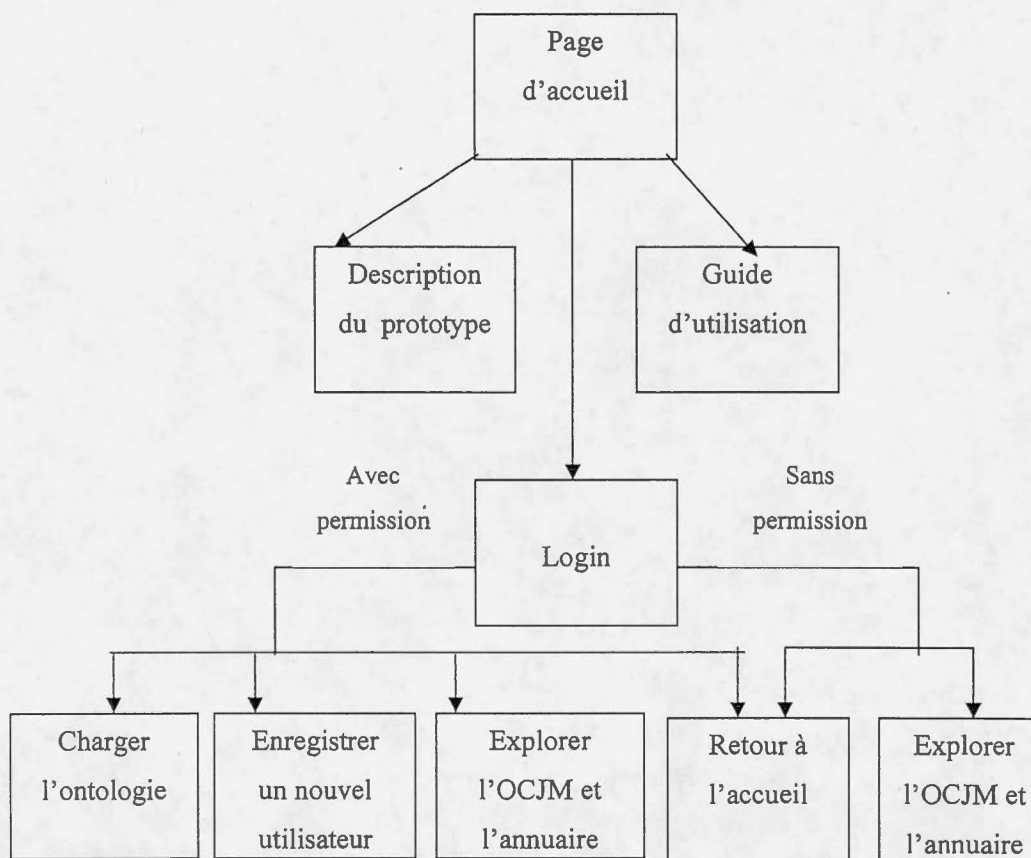


Figure 7.3 : Schéma général du site

Quand l'utilisateur se connecte et choisit l'option « explorer OCJM et annuaire » (fig.7.3), il aura la possibilité d'explorer les différentes sections de l'ontologie OCJM en consultant les définitions et les explications correspondantes sauvegardées dans l'annuaire électronique.

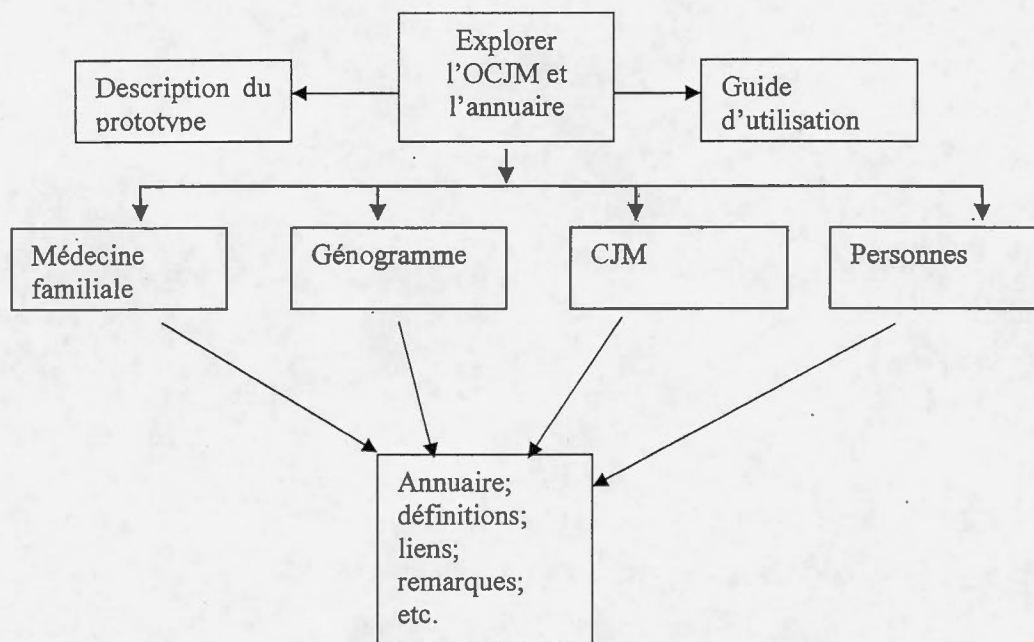


Figure 7.4 : Navigation dans le site

Les sections suivantes présentent des scénarios d'utilisation du prototype à l'aide d'exemples détaillés.

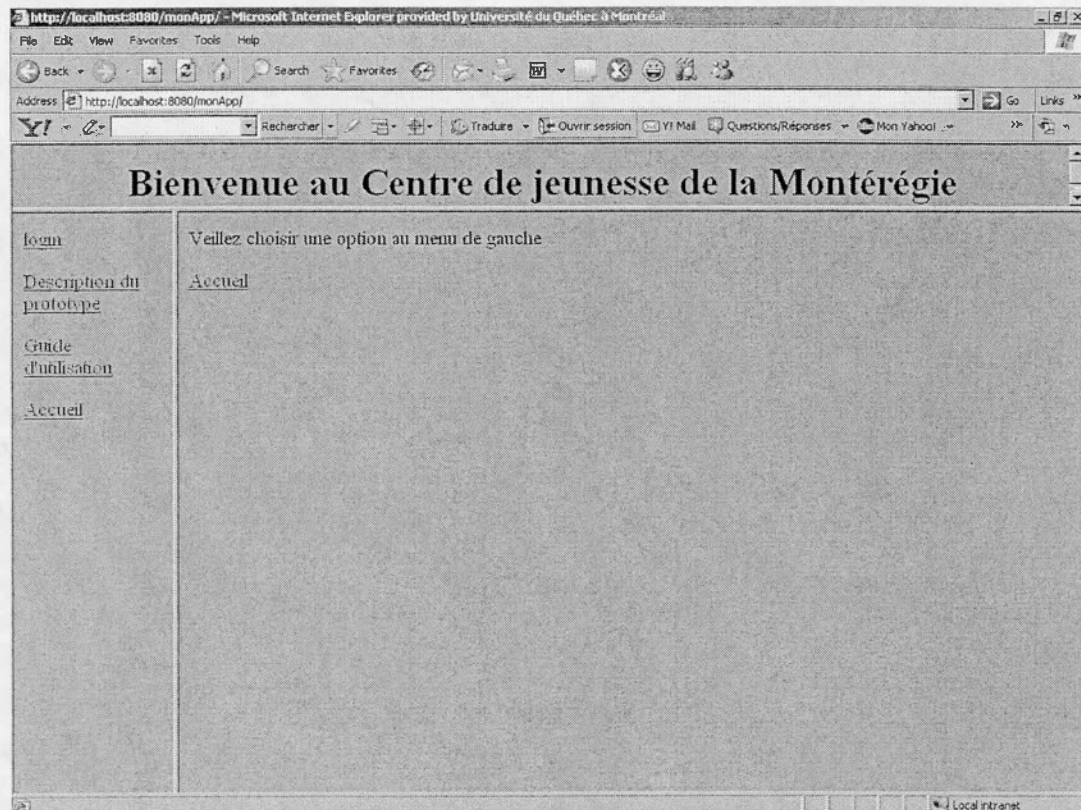


Figure 7.5 : La page d'accueil

En se connectant au site Web de l'application, la page d'accueil (fig. 7.5) apparaîtra. Elle offre à l'utilisateur trois options :

- consultation du document décrivant le prototype;
- consultation du guide d'utilisation du prototype;
- connexion (*login*) de l'utilisateur s'il est déjà inscrit dans la base de données *utilisateurs* pour commencer la navigation dans le site. Dans le cas contraire, il devrait contacter la personne responsable pour être inscrit.

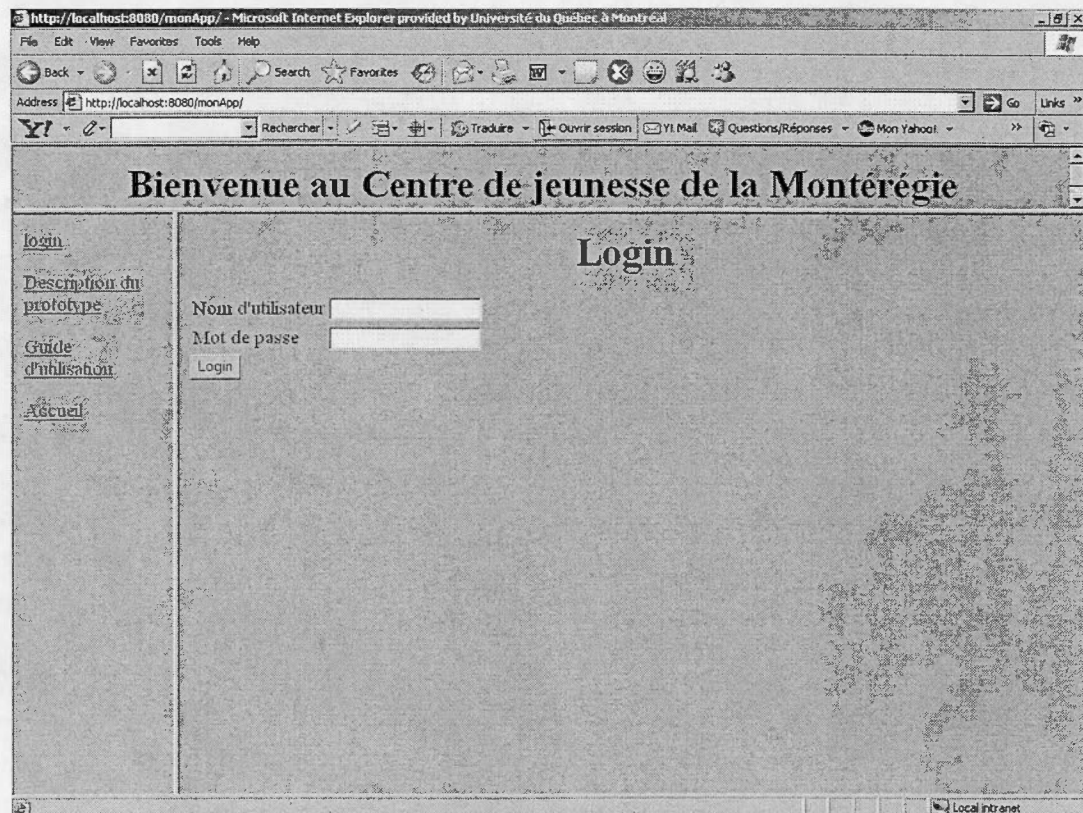


Figure 7.6 : La page « Login »

Si l'utilisateur avait choisi l'option *login* à la page d'accueil, alors cette page apparaîtra.

L'utilisateur est invité à entrer son nom d'utilisateur et son mot de passe inscrits dans la BD *utilisateurs*.

Après la vérification et la validation des entrées, la page de la figure (7.7) apparaîtra.

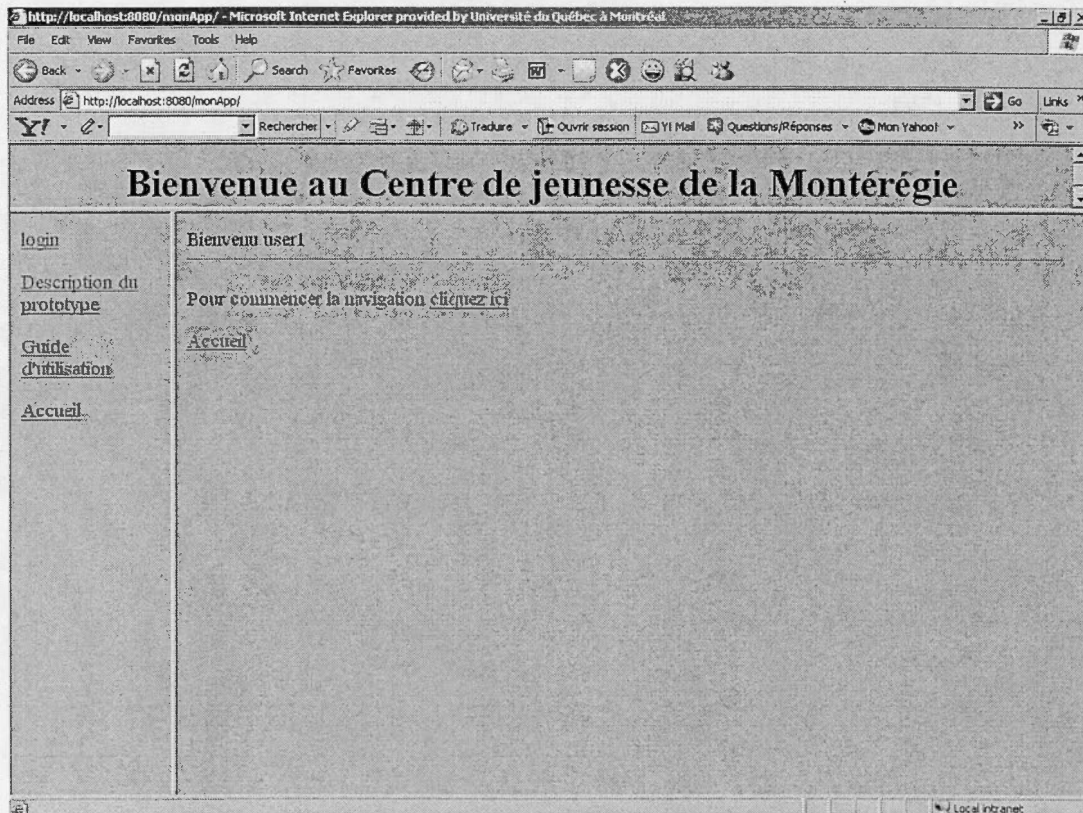


Figure 7.7 : Un utilisateur inscrit

Cette page (fig. 7.7) est accessible à tout utilisateur muni d'un nom d'utilisateur et d'un mot de passe enregistrés dans la BD *utilisateurs*. Elle permet de commencer la navigation dans le site et consulter les connaissances stockées dans la base des connaissances ou dans l'annuaire électronique.

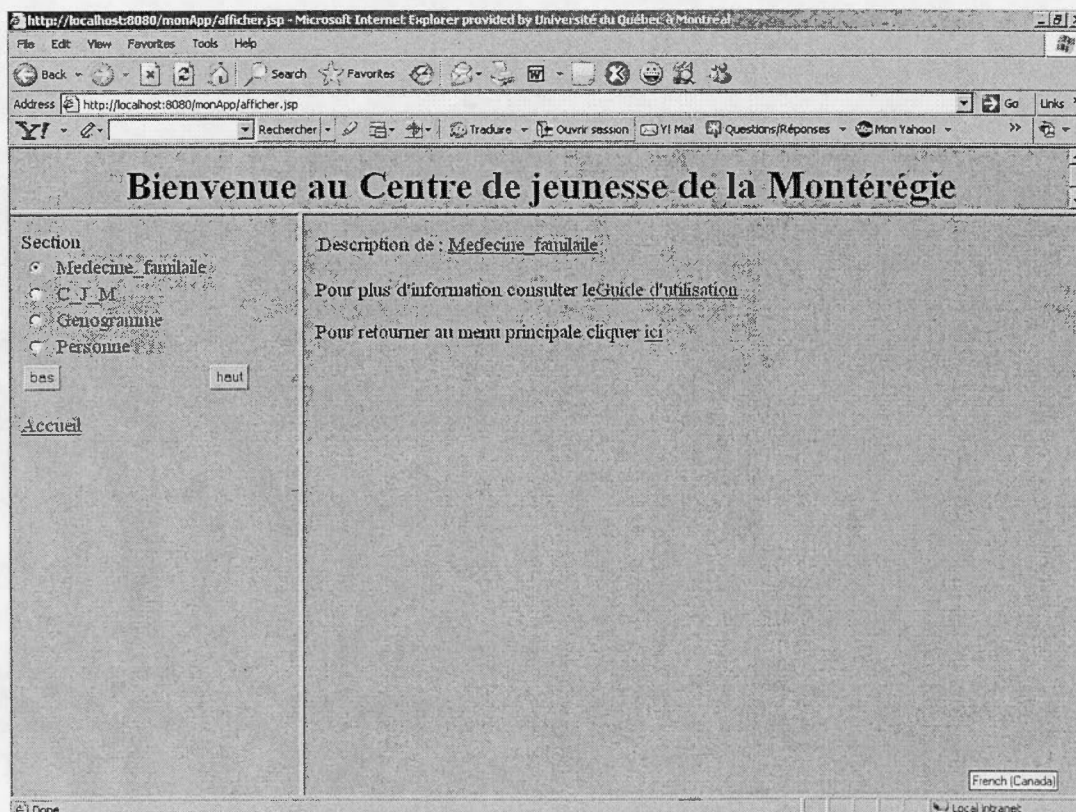


Figure 7.8 : Début de la navigation

La page de la figure (7.8) affiche, dans le cadre de gauche, les quatre principales parties de l'ontologie OCJM. Quand l'utilisateur choisit une section en cochant le bouton radio correspondant, trois options lui sont offertes :

- Il pourra appuyer sur le bouton « bas » pour visualiser les sous-sections (ou sous-classes) de la section choisie (fig. 7.9).
- Il pourra également appuyer sur le bouton « haut » pour consulter les classes mères (classes supérieures) de la section choisie.
- Il pourra cliquer sur le lien hypertexte à droite pour consulter la fiche descriptive de la section choisie, enregistrée dans l'annuaire électronique (fig. 7.10).

De plus, il peut à tout moment consulter le guide d'utilisation ou le document de description du prototype, il peut aussi retourner à la page d'accueil.

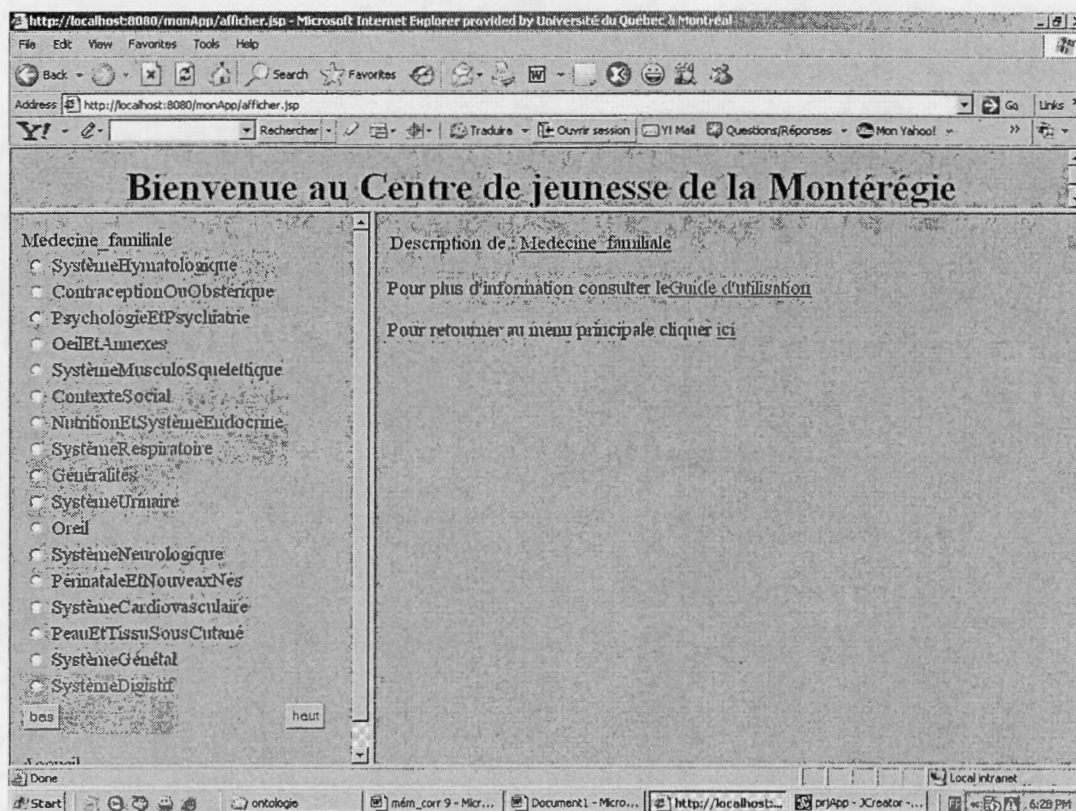


Figure 7.9 : Exemple de sous-classes

Cette page s'affiche quand l'utilisateur a choisi, dans la page précédente, la section « Médecine familiale » et a appuyé sur le bouton « bas ». Elle présente les différents sous-classes de la médecine familiale selon l'ontologie OCJM. Elle offre également à l'utilisateur les trois options présentées à la page précédente (fig.7.8).

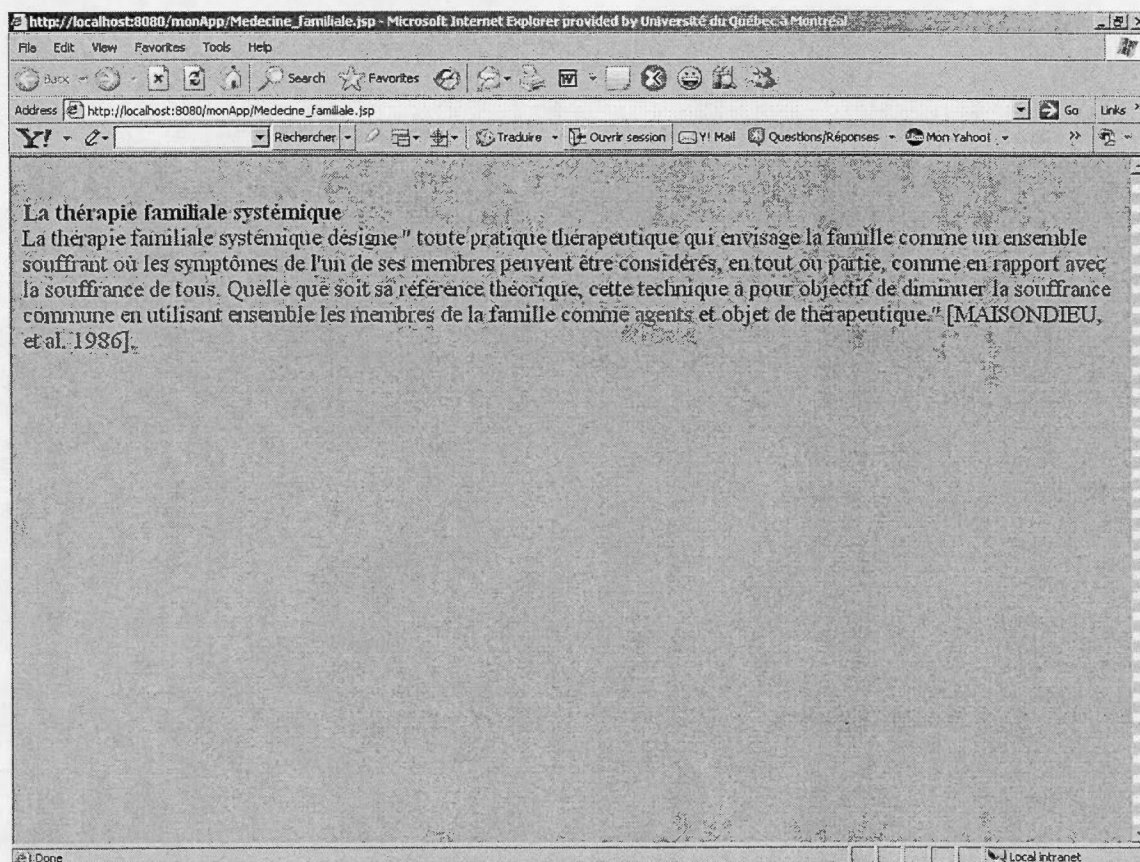


Figure 7.10 : Exemple de consultation de l'annuaire

Cette page s'affichera quant l'utilisateur aura cliqué sur le lien hypertexte « Médecine familiale » du cadre de droite de la page présentée à la figure 7.8. Elle s'affichera dans une fenêtre indépendante afin de ne pas perturber l'utilisateur dans sa navigation d'apprentissage. Évidemment, l'utilisateur peut réduire, agrandir ou fermer cette fenêtre à tout moment. Cette page fait partie de l'annuaire électronique.

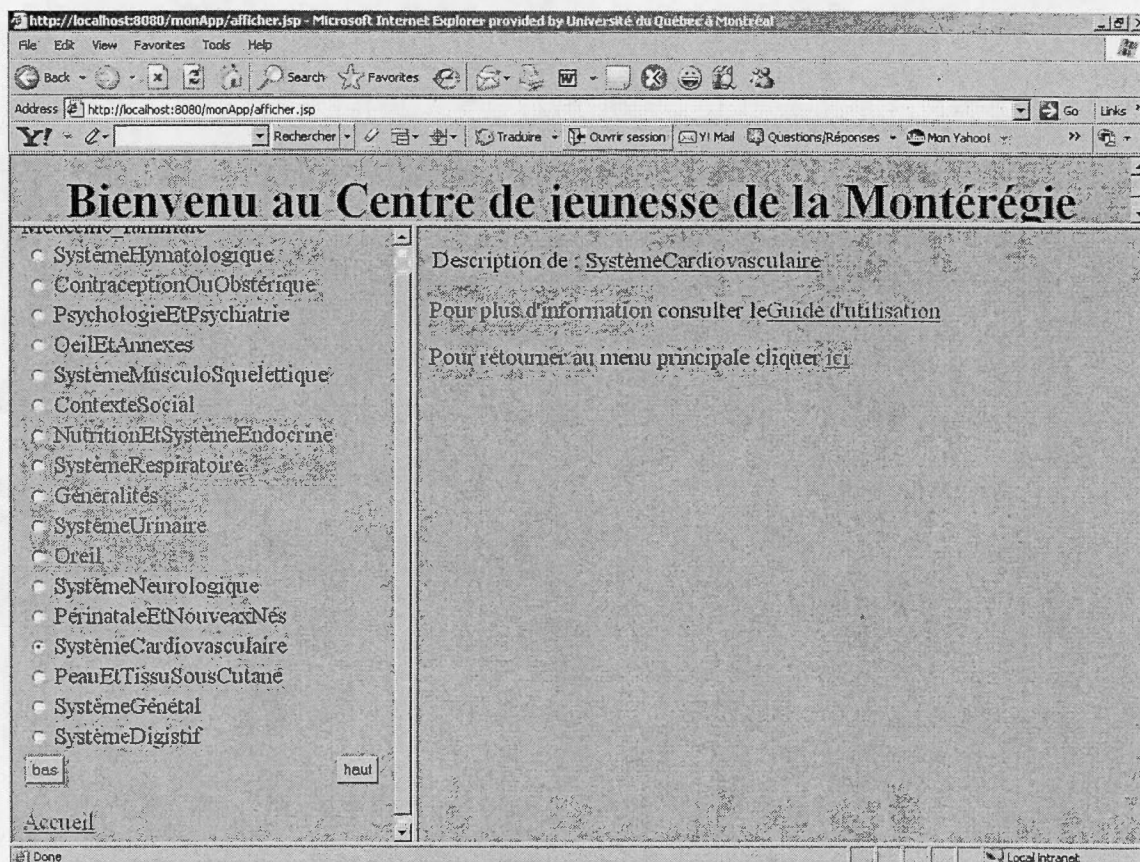


Figure 7.11 : Choix d'une sous-classe

Cette figure présente les sous-catégories de la médecine familiale à gauche et un lien vers la section « Système Cardiovasculaire » à droite. Ce lien est rendu disponible après que l'utilisateur ait coché le bouton radio correspondant dans le cadre de gauche.

L'utilisateur a toujours la possibilité de continuer l'exploration de l'ontologie (« bas » ou « haut ») ou de consulter la page de « Système Cardiovasculaire » dans l'annuaire électronique.

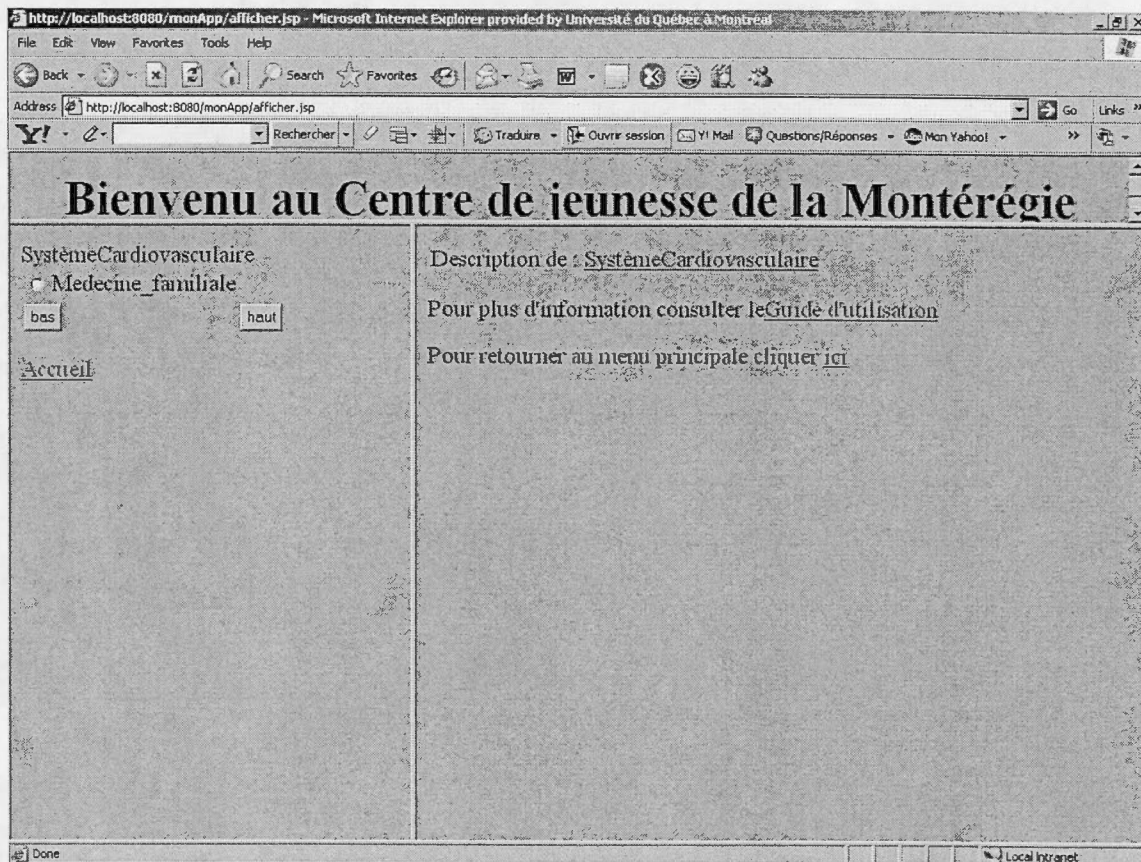


Figure 7.12 : Retour à la classe mère

Après avoir choisi le bouton radio « SystèmeCardioVasculaire » dans la page précédente, si l'utilisateur avait appuyé sur le bouton « haut », cette page (fig. 7.12) serait affichée. Elle présente la classe supérieure « Médecine familiale » de la classe choisie « SystèmeCardioVasculaire ».

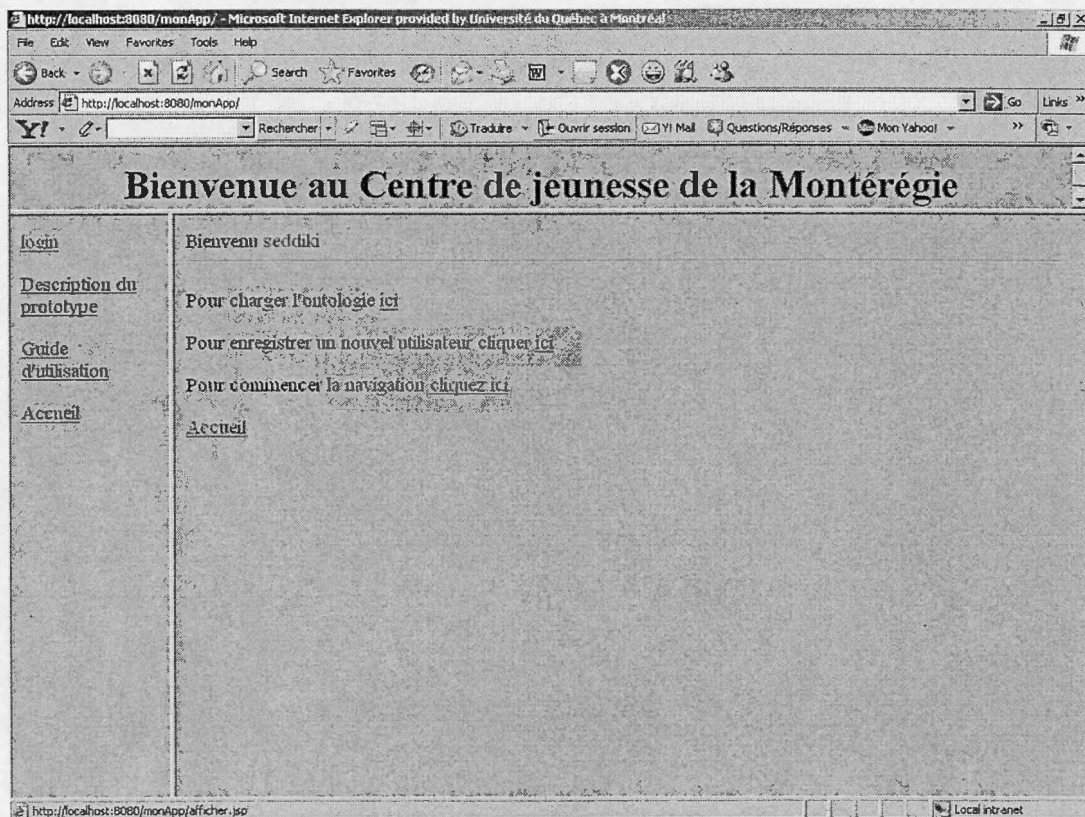


Figure 7.13 : Utilisateur avec privilège

Si l'utilisateur est enregistré dans la base de données et qu'il dispose de permissions requises, cette page (fig. 7.13) lui sera affichée après son *login*. Alors, il aura en plus des possibilités présentées auparavant, les deux options suivantes :

- Charger l'ontologie dans la base de données.
- Enregistrer un nouvel utilisateur.

S'il choisit d'enregistrer un nouvel utilisateur, alors la page suivante sera affichée (fig. 7.14). S'il choisit de charger l'ontologie dans la BD alors la page de la figure (fig. 7.15) sera affichée.

The screenshot shows a Microsoft Internet Explorer window with the address bar displaying `http://localhost:8080/monApp/`. The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The address bar also shows a search bar with the text "Rechercher" and a "Go" button. The main content area has a header "Bienvenue au Centre de jeunesse de la Montérégie". Below this, there is a sidebar on the left with links: [login](#), [Description du prototype](#), [Guide d'utilisation](#), and [Accueil](#). The main content area is titled "Enregistrement d'un nouvel utilisateur" and contains a registration form with the following fields: "Nom d'utilisateur" (containing "cestmoi"), "Nom" (containing "Tremblay"), "Prénom" (containing "Michel"), and "Mot de passe" (containing "*****"). There is an "envoyer" button below the password field. The status bar at the bottom shows "Done" and "Local intranet".

http://localhost:8080/monApp/ - Microsoft Internet Explorer provided by Université du Québec à Montréal

File Edit View Favorites Tools Help

Back Search Favorites

Address http://localhost:8080/monApp/ Go Links

Y! Rechercher Traduire Ouvrir session Y! Mail Questions/Réponses Mon Yahoo!

Bienvenue au Centre de jeunesse de la Montérégie

[login](#)
[Description du prototype](#)
[Guide d'utilisation](#)
[Accueil](#)

Enregistrement d'un nouvel utilisateur

Nom d'utilisateur : cestmoi
Nom : Tremblay
Prénom : Michel
Mot de passe : *****
envoyer

Done Local intranet

Figure 7.14 : Enregistrement d'un nouvel utilisateur

Cette page (fig. 7.14) permet d'enregistrer un nouvel utilisateur. Seules les personnes autorisées, qui ont accès à la base de données, ont la possibilité d'enregistrer un nouvel utilisateur.

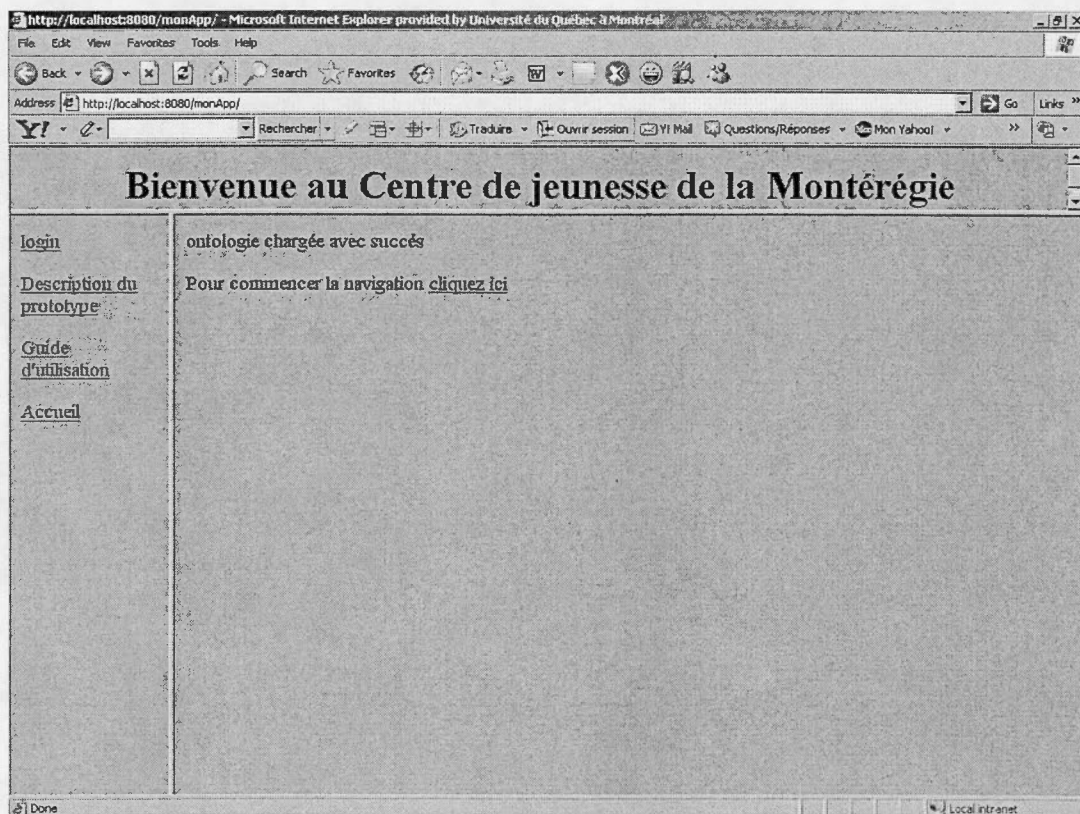


Figure 7.15 : Chargement d'ontologie avec succès

Cette page (fig. 7.15) affiche un message qui affirme que l'ontologie a été chargée avec succès. Par conséquent, elle offre à l'utilisateur la possibilité de commencer sa navigation dans le site.

Dans ce chapitre nous avons présenté l'architecture du SGC, les technologies utilisées ainsi que le guide d'utilisation du prototype. Au prochain chapitre nous présentons la conclusion générale du mémoire.

CONCLUSION

Cette conclusion rappelle les activités réalisées dans ce projet de recherche, elle présente les contributions du mémoire et ses limites et propose quelques perspectives de recherche sur le sujet.

Dans le cadre de cette recherche, nous avons appliqué une approche à base d'ontologie pour résoudre certaines problématiques liées à la gestion des connaissances. Tout en répondant aux trois questions que nous nous sommes posées au chapitre 1, nous avons réalisé les activités suivantes :

- 1 Identification des principales connaissances à gérer afin de résoudre les problématiques soulevées. Nous avons regroupé ces connaissances en quatre sections constituant l'ontologie OCJM (chap. 6). Cet élément est la réponse à la première question de recherche : « Quelles sont les connaissances nécessaires (cruciales) que chaque employé devrait connaître et auxquelles il aurait accès rapidement et facilement afin d'éviter les confusions et de pouvoir communiquer (au sens large) efficacement avec ses collègues? ».
- 2 Choix d'une méthodologie de développement d'ontologie à savoir la méthodologie de Natalia et McGuinness (2003) inspirée du génie cognitif. Ce choix est fait après la comparaison des principales méthodologies existantes.
- 3 Construction de l'ontologie OCJM qui est constituée de quatre principales parties : la médecine familiale, le CJM, les personnes et le génogramme. Cette ontologie constitue une bonne façon de représenter les connaissances cruciales identifiées. Ceci est la réponse à notre deuxième question « Quelle serait la meilleure façon de représenter ces connaissances cruciales afin de les conserver et d'en faciliter l'accès? ».

- 4 Choix du langage de formalisation de l'ontologie, il s'agit du langage OWL. Ce choix a été évident pour nous compte tenu des points suivants :
 - OWL offre plus de fonctionnalités que les autres langages d'ontologie (chap. 5);
 - OWL est le dernier standard recommandé par l'organisation W3C;
 - OWL permettra à notre application d'être compatible avec les nouvelles technologies du Web sémantique.
- 5 Conception et réalisation du prototype qui nous ont amené à :
 - Créer une base de donnée MySQL;
 - Développer un site Web entièrement dynamique, ce qui facilite sa maintenance et son extension;
 - Développer une application pour charger l'ontologie dans la base de données;
 - Utiliser les technologies les plus efficaces telles que l'API Jena et les JSP.

Ce système (prototype) est la réponse pratique à la troisième question de recherche qui est « Comment pourrait on transférer ces connaissances et les partager entre les différents employés? ». En effet, ce système permet le transfert et le partage des connaissances stockées dans l'ontologie entre les employés.

Les contributions

Nous estimons que les contributions de cette recherche sont de deux types : théoriques (académiques) et pratiques (managériales).

Les contributions théoriques

Tout d'abord, la revue de littérature sur les ontologies constitue selon nous un apport théorique utile pour le domaine d'application. En effet, elle propose à l'aide de multiples comparaisons, une étude synthétique sur les ontologies et leurs utilisations dans la gestion des connaissances. Nous avons également proposé un cadre conceptuel de base sur le rôle des ontologies. Ce cadre conceptuel constitue un support théorique pour les différents courants travaillant sur les ontologies, notamment le courant d'intelligence artificielle et de systèmes d'information.

Par ailleurs, nous espérons que le lecteur de ce mémoire aura une idée complète, claire et structurée sur les différents aspects des ontologies (typologie, méthodologie, formalisation...etc.).

Les contributions pratiques et managériales

D'un point de vue managérial, nous avons montré en quoi notre recherche peut aider les gestionnaires à résoudre certains problèmes liés à la GC. En effet, le CJM peut utiliser le SGC que nous avons développé pour résoudre les problématiques liées à la communication, la recherche d'information et l'intégration des nouveaux employés.

Par ailleurs, cet apport managérial peut être bénéfique pour d'autres organisations, car il permet de prendre conscience des avantages des ontologies dans la gestion des connaissances. De plus, l'approche que nous avons suivie est applicable dans d'autres organisations spécialisées dans d'autres domaines.

Les limites de la recherche

Les principales contributions que nous avons identifiées présentent aussi des limites. En effet, l'ontologie OCJM devrait être enrichie en y ajoutant d'autres sections et d'autres concepts aux quatre sections existants. De plus, l'annuaire électronique devrait être complété et amélioré en y ajoutant d'autres pages et d'autres informations utiles.

Par ailleurs, l'interface de l'application devrait être améliorée car le long de ce projet, nous avons mis plus d'emphasis sur les fonctionnalités et la performance de l'application.

Perspectives de recherche

Il reste beaucoup de recherche à faire dans le domaine des approches de gestion des connaissances basées sur les ontologies. Ces recherches portent principalement sur la proposition d'un cadre conceptuel pour les rôles des ontologies et la proposition d'une méthodologie standard pour la construction des ontologies.

APPENDICE A

Exemple de termes utilisés au CJM

Enfant	Antécédents personnels,	Relation d'union
Jeune	Médicaux et psychiatrique	Relation affective
	Histoire sociale	Relation de descendance
Père	Occupation, travail à risque	Relation d'ascendance
Mère	Famille	Relation de jumeau
Parent	Sport et loisirs	Adoption
Grand parant	Habitude de vie	Les symboles
Grand père maternel	Alimentation	Situation familiale
Grande mère maternelle	Tabac	Patterns
Grand père paternel	R-OH, caféine, drogue	CLSC
Grand mère paternel	Voyage	Cliniques
Conjoint	Médication	Médecin
Conjointe	Risque élevé	Thérapeute
Frère	Génogramme	Psychologue
Sœur	Génogramme informatisé	Famille complexe
Fille	Chirurgie	Famille recomposée
Fils	Allergie	Parents biologiques
Neveu	Vaccination	Parents légaux
Nièce	Les professionnels de la santé	Couple homosexuel
Petit-fils	Travailleurs sociaux	Enfant adopté
Petite-fille	Médecin de famille	Vue systémique
Petit-neveu	Infirmier	L'histoire médicale de la famille
Petite-nièce	Agent d'administration	Maladies héréditaires
Amis	Thérapie familiale	Mal. psychiatrique
Décédé	Recherche clinique	Allergie Mal. Cardiaque
Année de naissance	héritage	Mal. neurologique
Lien de parenté	Théorie des systèmes familiaux	Patient
Arbre généalogique	Arbre généalogique	Client
Antécédents familiaux	Individu	Consultation
Glaucome	structure familiale	Centre de jeunesse
Tuberculose	Avortement provoqué	Service
Diabète	Avortement spontané	Contexte social
Cancer	Mort-né	Contexte familial
Goutte	Grossesse	Événement
hyperlipidémie	Relation	Dossier médical
HTA	Protection des jeunes	Modèles familiaux
État émotionnel	Prévention	Condition de vie
État physique	Signalement	Trouble psychologique
Hôpital	Urgence sociale	Garde partagée
Hébergement	Orientation des jeunes	...etc.
réadaptation		

Tableau A.1 : Termes utilisés au CJM

APPENDICE B

Exemple : la classe « Génogramme »

Tableau B.1 : La classe « Génogramme »

Classe	Description
Génogramme	Le génogramme est une représentation graphique de la structure familiale d'un individu analogue à un arbre généalogique
Relation [Génogramme]	Relation familiale ou émotionnelle
Relation familiale [Relation]	Relation de couple ou de parenté
Relation émotionnelle [Relation]	Rupture Apathique / Indifférence Pauvre/ Distante <i>Plain</i> / Normal Amitié / proche Intimité / très proche Hostile/ Conflictuelle Fondue <i>Fused</i> Distante et Hostile Proche et hostile Fondue et hostile Violence Distante et violence Proche et violence Fondue et violence Abus Abus physique Abus émotionnel Abus sexuel Négligence Obsession <i>focused on</i> Jamais rencontrée

Relation de couple [Relation familiale]	Mariage Séparation de fait Séparation légale Divorce Nullité Veuf/veuve Engagement Engagement et cohabitation Cohabitation légale Cohabitation légale et séparation de fait Cohabitation légale et séparation légale Cohabitation légale et décès Cohabitation Cohabitation et séparation Cohabitation et décès <i>Non-sentimental cohabitation</i> <i>Dating/Sporadic relationship</i> <i>Dating and separation</i> Temporaire / <i>One night stand</i> Viole /relation de force Autres relation
Relation de parenté [Relation familiale]	Père / Mère Frère /Sœur Fille/ Fils Grand père maternel Grande mère maternelle Grand père paternel Grand mère paternel Oncle côté maternel Oncle côté paternel Tante côté paternel Tante côté maternel Cousine côté maternel Cousine côté paternel Cousin côté maternel Cousin côté paternel Neveu/ Nièce Petit-fils Petite-fille Petit-neveu Petite-nièce

APPENDICE C

Exemple : la classe « Personne »

Tableau C.1 : La classe « Personne »

Classe	Description	Propriété
Personne	Une personne est un patient ou un employé	Nom Prénom sexe
Patient [Personne]	Un patient est une personne qui reçoit un service au CJM	Age
Employé [Personne]	Un employé appartient au personnel de santé ou au personnel administratif	Poste Bureau Tel. courriel
PersonnelSanté [Employé]	Médecin de famille, psychologue, travailleur social...	
MédecinDeFamille [PersonnelSanté]		
Psychologue [PersonnelSanté]		
Infirmier [PersonnelSanté]		
PersonnelAdministratif [Employé]		
TravailleurSocial [Employé]		

APPENDICE D

UNE PARTIE DU FICHIER « OCJM.owl »

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns="http://www.owl-ontologies.com/OCJM.owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.owl-ontologies.com/OCJM.owl">
  <owl:Ontology rdf:about="">
  <rdfs:comment >
    Ontologie du centre de jeunesse de la Montérégie
  </rdfs:comment >
  <rdfs:label> OCJM </rdfs:label>

  <!-- Les definitions des classes -->

  <owl:Class rdf:about="#Section">
    <rdfs:comment >
      C'est la classe principale de l'ontologie OCJM, elle contient les quatre principales sections
      (parties)
    </rdfs:comment>
  </owl:Class>

  <owl:Class rdf:ID="C_J_M">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#Section"/>
```



```

    </rdfs:subClassOf>
    <rdfs:comment >Cette classe décrit le centre de jeunesse de la Montérégie, son organisation,
        ses services et ses département
    </owl:Class>
    <owl:Class rdf:ID="Medecine_familiale">
        <rdfs:subClassOf>
            <owl:Class rdf:ID="Section"/>
        </rdfs:subClassOf>
        <rdfs:comment >
            Cette classe décrit les différentes branches de la médecine familiale
        </rdfs:comment>
    </owl:Class>

    <owl:Class rdf:ID="Personne">
        <rdfs:comment >
            Une personne est un client (patient) ou un employé du CJM
        </rdfs:comment>
        <rdfs:subClassOf>

    <owl:Class rdf:about="#Section"/>
        </rdfs:subClassOf>
    </owl:Class>

    <owl:Class rdf:about="#Genogramme">
        <rdfs:subClassOf rdf:resource="#Section"/>
        <rdfs:comment >
            Cette classe décrit le génogramme , l'outil utilisé par les médecins de famille
        </rdfs:comment>
    </owl:Class>

```

```

</owl:Class>
<owl:Class rdf:about="#Personnel">
  <rdfs:comment > un Employé du CJM</rdfs:comment>

  <rdfs:subClassOf rdf:resource="#Personne"/>
</owl:Class>

<owl:Class rdf:ID="Client">
  <rdfs:subClassOf rdf:resource="#Personne"/>
  <rdfs:comment >un patient du CJM</rdfs:comment>
</owl:Class>

<owl:Class rdf:ID="Maladie">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Genogramme"/>
  </rdfs:subClassOf>
  <rdfs:comment>
    Comme le tuberculose, le diabète...</rdfs:comment>
</owl:Class>

<owl:Class rdf:about="#Relation">
  <rdfs:subClassOf rdf:resource="#Genogramme"/>
</owl:Class>

<owl:Class rdf:ID="Antécédant_familiale">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Genogramme"/>
  </rdfs:subClassOf>
  <rdfs:comment >comme l'alcoolisme, la violence...</rdfs:comment>
</owl:Class>

```

```

<owl:Class rdf:about="#Service">
  <rdfs:subClassOf rdf:resource="#C_J_M"/>
  <rdfs:comment >Les services offerts par le CJM</rdfs:comment>
</owl:Class>

```

```

<owl:Class rdf:ID="S_Adoption_québécoise_et_internationale">
  <rdfs:subClassOf rdf:resource="#Service"/>
</owl:Class>

```

```

<owl:Class rdf:ID="S_Réadaptation">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Service"/>
  </rdfs:subClassOf>
</owl:Class>

```

```

<owl:Class rdf:ID="S_Psychosocial">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Service"/>
  </rdfs:subClassOf>
</owl:Class>

```

```

<owl:Class rdf:about="#S_Protection">
  <rdfs:comment >le Service de protection de la jeunesse : la réception des signalement et
    urgence, l'évaluation / orientation et la révision
  </rdfs:comment>
  <rdfs:subClassOf rdf:resource="#Service"/>
</owl:Class>

```

```

<owl:Class rdf:ID="Réception_de_signalement">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="S_Protection"/>

```

```

    </rdfs:subClassOf>
  </owl:Class>

  <owl:Class rdf:ID="évaluation">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#S_Protection"/>
    </rdfs:subClassOf>
  </owl:Class>

  <owl:Class rdf:ID="Révision">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#S_Protection"/>
    </rdfs:subClassOf>
  </owl:Class>

  <owl:Class rdf:about="#S_Hébergement">
    <rdfs:subClassOf rdf:resource="#Service"/>
  </owl:Class>

  <owl:Class rdf:ID="Type_communautaire">
    <rdfs:subClassOf>
      <owl:Class rdf:ID="S_Hébergement"/>
    </rdfs:subClassOf>
  </owl:Class>

  <owl:Class rdf:ID="F_accueil_spécifique">
    <rdfs:subClassOf>
      <owl:Class rdf:about="#S_Hébergement"/>
    </rdfs:subClassOf>
  </owl:Class>

```

```

<owl:Class rdf:ID="F_accueil_régulière">
  <rdfs:subClassOf>
<owl:Class rdf:about="#S_Hébergement"/>

  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="Relation_émotionnelle">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Relation"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:about="#Relation_familiale">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Relation"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="Relation_de_couple">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Relation_familiale"/>
  </rdfs:subClassOf>
</owl:Class>

<owl:Class rdf:ID="Relation_de_parenté">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Relation_familiale"/>
  </rdfs:subClassOf>
</owl:Class>

```



```

<owl:Class rdf:ID="Personnel_de_la_santé">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Personnel"/>
  </rdfs:subClassOf>
</owl:Class>

```

```

<owl:Class rdf:ID="Personnel_administratif">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Personnel"/>
  </rdfs:subClassOf>
</owl:Class>

```

```

<owl:Class rdf:ID="Medecin_de_famille">
  <rdfs:subClassOf rdf:resource="#Personnel_de_la_santé"/>
</owl:Class>

```

```

<owl:Class rdf:ID="Psychologue">
  <rdfs:subClassOf rdf:resource="#Personnel_de_la_santé"/>
</owl:Class>

```

```

<owl:Class rdf:ID="Infirmier">
  <rdfs:subClassOf rdf:resource="#Personnel_de_la_santé"/>
</owl:Class>

```

```

<owl:Class rdf:ID="Travailleur_social">
  <rdfs:subClassOf rdf:resource="#Personnel_de_la_santé"/>
</owl:Class>

```

```

<owl:Class rdf:ID="Généralités">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
</owl:Class>

```

```

<owl:Class rdf:ID="SystèmeHymatologique">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
</owl:Class>

<owl:Class rdf:ID="PsychologieEtPsychiatrie">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
</owl:Class>

<owl:Class rdf:ID="ContraceptionOuObstétrique">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
  <rdfs:comment >Contraception_Obstétrique</rdfs:comment>
</owl:Class>

<owl:Class rdf:ID="OeilEtAnnexes">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
</owl:Class>

<owl:Class rdf:ID="ContexteSocial">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
</owl:Class>

<owl:Class rdf:ID="SystèmeMusculoSquelettique">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
  <rdfs:comment >
    Système Musculo-Squelettique
  </rdfs:comment>
</owl:Class>

<owl:Class rdf:ID="NutritionEtSystèmeEndocrine">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>

```

```
</owl:Class>
```

```
<owl:Class rdf:ID="SystèmeRespiratoire">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="SystèmeUrinaire">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="Oreil">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="SystèmeNeurologique">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="PérinataleEtNouveauxNés">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="SystèmeCardiovasculaire">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:ID="PeauEtTissuSousCutané">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
</owl:Class>
```

```
<owl:Class rdf:about="#SystèmeGénéral">
```

```
<rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
```

```
<rdfs:comment >Système_Général </rdfs:comment>
</owl:Class>

<owl:Class rdf:ID="SystèmeDigistif">
  <rdfs:subClassOf rdf:resource="#Medecine_familiale"/>
</owl:Class>

<owl:Class rdf:ID="S_G_Homme">
  <rdfs:subClassOf rdf:resource="#SystèmeGénéral"/>
</owl:Class>

<owl:Class rdf:ID="S_G_Femme">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="SystèmeGénéral"/>
  </rdfs:subClassOf>
</owl:Class>
```

APPENDICE E

LES PRINCIPALES CLASSES JAVA

```

/*****
*
* Auteur      Seddiki karim
* Courriel    karim_sedca@yahoo.ca
* Nom de fichier DbLien.java
* Date de création mai 2006
* Description  Ce composant JavaBean permet de créer un « pont de connection » vers
                la BD MySQL
*
*****/

// Package
//////////
package mesClasses ;
import java.io.Serializable ;
import java.sql.*;
public class DbLien implements java.io.Serializable{

//définition des propriétés
private Statement lien =null;

//Méthodes d'accès en lecture
public Statement getLien(Connection cnx){
    if(construireStatement(cnx)){
        return lien;
    }
}
```



```
        {  
            return null;  
        }  
    }  
  
    //Méthode utilitaires  
    private boolean construireStatement (Connection cnx) {  
        boolean statusStatement= false;  
  
        try {  
            lien=cnx.createStatement();  
            statusStatement=true;  
        } catch (Exception e){  
            statusStatement=false;  
            System.out.println("Impossible de créer un 'Statement'");  
        }  
        return statusStatement;  
    }  
}
```

```

/*****
*
* Classe DBConnexion permet de se connecter à la base de données
*
*****/

public class DBConnexion implements java.io.Serializable{

    // Variables
    private String login;
    private String password;
    private String hostname;
    private String port;
    private String nomDeLaBase;
    private Connection cnx;
    private String urlJdbc;

    //Méthodes d'accès en écriture
    public void setLogin(String val){
        login=val;
    }

    public void setPassword(String val){
        password=val;
    }

    public void setHostname(String val){
        hostname=val;
    }

    public void setPort(String val){
        port=val;
    }
}

```

```

        public void setNomDeLaBase(String val){
            nomDeLaBase=val;
        }

//Méthodes d'accès en lecture
public Connection getCnx(){
    if (etablirConnexion()) {return cnx;
    } else {
        return null;
    }
}

// Méthodes utilitaires (métier)

private String construireUrlJdbc(){

    urlJdbc="jdbc:mysql://" + hostname+":"+ port+ "/" + nomDeLaBase;
    urlJdbc=urlJdbc+ "?user=" +login + "&password=" +password;
    return urlJdbc;
}

// Fontion : etablirConnexion

private boolean etablirConnexion(){
    boolean statusConnexion=false;
    try {
        Class.forName ("com.mysql.jdbc.Driver");
        urlJdbc= construireUrlJdbc();
        cnx=DriverManager.getConnection(urlJdbc);
        statusConnexion=true;
    }
}

```

```

    } catch(Exception e){statusConnexion=false;
        System.out.println("connexion impossibleeeee");
    }

    return statusConnexion;
}
}

```

```

//***** Classe Login *****

```

```

public class Login implements java.io.Serializable {

    // Définition des variables (propriétés)
    private String nomUtilisateur;
    private String motPasse ;
    private String nom;
    private String prenom ;

    // Définition des methodes d'accès en écriture

    // Le nom de l'utilisateur
    public void setNomUtilisateur (String valeur) {
        nomUtilisateur =valeur;
    }

    // Le mot de passe de l'utilisateur
    public void setMotPasse (String valeur) {
        motPasse =valeur;
    }
}

```

```
        // Le nom
public void setNom (String valeur) {
    nom =valeur;
}
// Le prenom
public void setPrenom (String valeur) {
    prenom =valeur;
}
```

```
// Définition des methodes d'accès en lecture
```

```
// Le nom de l'utilisateur
public String getNomUtilisateur () {
    return nomUtilisateur;
}
```

```
// Le mot de passe
public String getMotPasse () {
    return motPasse;
}
```

```
        // Le nom
public String getNom () {
    return nom;
}
// Le prenom
public String getPrenom () {
    return prenom;
}
```

```
}
```



```

/*****
*
* Auteur      Seddiki karim
* Courriel    karim_sedca@yahoo.ca
* Nom de fichier  ChargerApp.java
* Date de création  mai 2006
* Description  Ce composant javaBeans permet de charger l'ontologie « OJCM.owl »
*              dans la base de données «BDOCJM »
*
*****/

// Package
//////////
package mesClasses;

// Imports
//////////
import java.util.*;
import com.hp.hpl.jena.db.*;
import com.hp.hpl.jena.ontology.*;
import com.hp.hpl.jena.rdf.model.*;
import java.sql.*;

public class ChargerApp implements java.io.Serializable {

    // Constantes
    //////////////////////////////////////
    public static final String DB_URL = "jdbc:mysql://localhost/BDOCJM";
    public static final String DB_USER = "root";
    public static final String DB_PASSWD = "root";

```

```

public static final String DB = "MySQL";
public static final String DB_DRIVER = "com.mysql.jdbc.Driver";

// Variables statiques
////////////////////////////////////

// paramètres de connection à la BD

private static String s_dbURL = DB_URL;
private static String s_dbUser = DB_USER;
private static String s_dbPw = DB_PASSWD;
private static String s_dbType = DB;
private static String s_dbDriver = DB_DRIVER;
private static String source="http://localhost:8080/monApp/OCJM.owl";

// Variables
////////////////////////////////////

private boolean drive =false ;
private ModelMaker dbMaker ;
private boolean loadDb;

// chargement de pilote

private boolean connection (){
    try {

        Class.forName (s_dbDriver);
        return true;

    } catch(Exception e){
        System.out.println("connexion impossibleeeee");
    }
}

```

```

return false; }
}

```

// La fonction getDrive() renvoie true si le driver à été chargé correctement, false sinon

```

public boolean getDrive(){
    drive=connection();
    return drive;
}

```

// chargement de l'ontologie

// la fonction getLoadDb() renvoie true si l'ontologie est chargée, false sinon

```

public boolean getLoadDb() {

    try{

        connection ();
        dbMaker=getDbMaker( );
        Model base = dbMaker.createModel(source);
        OntModel m = ModelFactory.createOntologyModel(
            getModelSpec( dbMaker ), base );
        m.read( source );
        loadDb=true;

    } catch(Exception e){loadDb= false; }

    return loadDb
    ;
}

```

```
// Création du model Jena
```

```
public OntModel listClasses( ) {
```

```
    Model base = dbMaker.createModel( source, false );
```

```
    OntModel m = ModelFactory.createOntologyModel( getModelSpec( dbMaker ), base );
```

```
    return m ;
```

```
}
```

```
// Connection à la BD
```

```
public ModelMaker getDbMaker( ) { try { IDBConnection conn = new DBConnection(
DB_URL, DB_USER, DB_PASSWD, DB );
```

```
    if (cleanDB) conn.cleanDB();
```

```
    dbMaker= ModelFactory.createModelRDBMaker( conn );
```

```
    } catch (Exception e){
```

```
        e.printStackTrace();
```

```
        dbMaker= null; }
```

```
return dbMaker;
```

```
}
```

```
// Choix de moteur d'inference OWL
```

```
public OntModelSpec getModelSpec( ModelMaker maker ) {
```

```
    OntModelSpec spec = new OntModelSpec( OntModelSpec.OWL_MEM );
```

```
    spec.setImportModelMaker( maker )
```

```
    return spec;
```

```
}}
```

APPENDICE F

LES PRINCIPALES PAGES JSP

/*

*

* Auteur Seddiki karim

* Courriel karim_sedca@yahoo.ca

* Nom de fichier ChargerApp.JSP

* Date de creation mai 2006

* Description Cette page JSP permet de charger l'ontologie « OCJM.owl »

* dans la base de données « BDOCJM » elle utilise le JavaBean
« ChargerApp »

*

*****/

<%@ page import="java.util.*" %>

<%@ page import="java.sql.Connection" %>

<%@ page import="java.sql.Statement" %>

<%@ page import="java.sql.ResultSet" %>

<%@ page import=" com.hp.hpl.jena.ontology.*" %>

<%@ page import=" com.hp.hpl.jena.db.*" %>

<%@ page import ="com.hp.hpl.jena.rdf.model.*"%>

<%! boolean load ;%>

<%! ModelMaker mMaker ;%>

<%! OntModel m ;%>

<jsp:useBean id="charApp" class ='mesClasses.ChargerApp' />


```
<%  
load = charApp.getLoadDb();  
if (load == false){  
    out.println(" impossible de charger l'ontologie ");  
}  
else {  
    out.println(" ontologie chargée avec succès");  
}  
out.println("<p>Pour commencer la navigation <a href='afficher.jsp' Target='_top' >  
cliquez ici</a> </p>");  
%>
```

```

/*****

```

```

*

```

```

* Auteur      Seddiki karim

```

```

* Courriel    karim_sedca@yahoo.ca

```

```

* Nom de fichier  ValiderLogin.JSP

```

```

* Date de creation  mai 2006

```

```

* Description    Cette page JSP permet de valider le login d'un utilisateur

```

```

*                dans la base de données utilisateurs

```

```

*                elle utilise le JavaBeans DBConnexion

```

```

*****/

```

```

<%@ page import="java.sql.Connection" %>

```

```

<%@ page import="java.sql.Statement" %>

```

```

<%@ page import="java.sql.ResultSet" %>

```

```

<jsp:useBean id="dbcnx" class='mesClasses.DBConnexion'>

```

```

<jsp:setProperty name="dbcnx" property="login" value="root" />

```

```

<jsp:setProperty name="dbcnx" property="password" value="root" />

```

```

<jsp:setProperty name="dbcnx" property="hostname" value="localhost" />

```

```

<jsp:setProperty name="dbcnx" property="port" value="3306" />

```

```

<jsp:setProperty name="dbcnx" property="nomDeLaBase" value="users" />

```

```

</jsp:useBean>
<jsp:useBean id="dblien" class='mesClasses.DbLien' />
<html>
<body>
<%
Connection cnx= dbcnx.getCnx();
if(cnx==null){
    out.println("impossible " + cnx);
}
else {
    //out.println("OK" + cnx);
}
%>
<%

Statement lien= dblien.getLien(cnx);
if(lien==null){
    out.println(" lien impossible " + lien );
}
else {
    //out.println("lien OK " + lien);
}
%>
<jsp:useBean id="user" class='mesClasses.Login'/>
<jsp:setProperty name="user" property="nomUtilisateur" param="nomUser"/>
<jsp:setProperty name="user" property="motPasse" param="Mot_de_passe"/>

```

```

<%
if (user.getNomUtilisateur() !=null && user.getMotPasse()!=null){
String commande2="select userName, passWord from tabusers where userName="

+ user.getNomUtilisateur() + " and passWord=""+
user.getMotPasse()+ """;
ResultSet rs2;
rs2=lien.executeQuery(commande2);
if (rs2.first()) {
String nomuser=rs2.getString("userName");
String mot_passe=rs2.getString("passWord");
out.println(" Bienvenu " + nomuser);
out.println(" <hr>");
if( nomuser.equals("seddiki") && mot_passe.equals("karim"))
{
out.println("<p>Pour charger l'ontologie <a href='charger_app.jsp'> ici</a> </p>");
out.println("<p>Pour enregistrer un nouvel utilisateur cliquer <a
href='enregistrement.jsp'> ici</a> </p>");
}
}
else {out.println(" Nom d'utilisateur ou mot de passe incorrect");
out.println("<p>Pour essayer à nouveau cliquer <a href='login.jsp' > ici</a> </p>");
}}
else {out.println(" Nom d'utilisateur ou mot de passe incorrect");
out.println("<p>Pour essayer à nouveau cliquer <a href='login.jsp'> ici</a> </p>");}
out.println("<p>Pour commencer la navigation <a href='afficher.jsp' Target='_top' >
cliquez ici</a> </p>")
}%>
</body>
<p><A href="index.jsp" Target='_top' >Accueil</A></p></html>

```

Bibliographie

- Alavi, Maryam et Dorothy E. Leidner. (1999). « Knowledge management systems: issues, challenges, and benefits ». *Communications of the AIS*, vol.1, no7 (février).
- Béland, Normand. (1993) « Le génogramme : une fenêtre ouverte sur la famille », *Le médecin du Québec*, Vol. 28.
- Bernaras, A., I. Laresgoiti, et J. Corera. « Building and reusing ontologies for electrical network applications ». In *Proceedings of the 12th ECAI*, pages 298–302, 1996
- Berners-Lee Tim, Hendler James et Lasilla Ora. (2001). « The Semantic Web », *Scientific American*, mai 2001. Consulté le 15 janvier 2006 sur le World Wide Web: <http://www.w3.org/2001/sw/>.
- Bernstein, Robert M., Gary R. Hollingworth et Gary S. Viner. (2001). « ENCODE-FM: Electronic Nomenclature and Classification Of Disorder and Encounters for Family Medicine », *Insite Family Medecine Inc.* version 3.0, (Decembre).
- Bertalanffy, Ludwig von. (1993). « Théorie générale des systèmes ». Trad. de l'anglais par Jean Benoist Chabrol. Paris : Dunod, 308 p.
- Borst, Willem Nico. (1997). « Construction of engineering ontologies for knowledge sharing and reuse ». Thèse de doctorat . *University of Twente, Netherlands*. Consulté le 1 février 2006 sur le World Wide Web : <http://doc.utwente.nl/17864>
- Bouteillier, J. (1999). « La ruée vers l'or gris ». *Neteconomie*. Disponible sur le World Wide Web: www.neteconomie.fr/dossiers/dossierkm.html
- Bowen, M. (1978) . « Family therapy in clinical practice ». New York, *Garden Press*.
- Chenel, Jean-Pierre. (1994) . « Le génogramme informatisé ». Mémoire de maîtrise en informatique de gestion, UQAM.
- Davenport, Thomas. (2003). « La gestion des connaissances: Évolution et perspective » CEFRIO, La gestion du savoir, Acte du colloque. Consulté le 15 février 2006 sur le World Wide Web: <http://www.cefrio.qc.ca/>.
- Dieng Rose, Olivier Corby, Alain Giboin, Myriam Ribi re. (1998). « Methods and Tools for Corporate Knowledge Management » Rapport de recherche de l'INRIA France. Consult  le 30 f vrier 2006 sur le World Wide Web <http://www.inria.fr/rrrt/rr-3485.html>

- Edgington, Theresa, Beomjin Choi, Katherine Henson, T.S. Raghu et Ajay Vinze. (2004) « Adopting Ontology to Facilitate Knowledge Sharing ». Communication of the ACM vol. 47. no.11 (novembre).
- Ermine, J-L. (2003). « La gestion des connaissances ». Paris : Hermes science : Lavoisier, 166 p.
- Ermine, J-L. (2001). « Ingénierie et capitalisation des connaissances ». Paris : Hermès : science publications. Chap. 4, p.65-102.
- Fernández, López M. (1999) « Overview Of Methodologies For Building Ontologies » Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5) Stockholm, Sweden, August 2.
- Gaëlle, Simon. (2001) « Capitaliser les connaissances par la mise en œuvre de mémoires d'entreprise techniques ». In Ingénierie et capitalisation des connaissances. P133-150. Paris : Hermès.
- GenoPro Inc(2006). Consulté le 25 février 2006 sur le World Wide Web: <http://www.genopro.com>.
- Gerson, Randy et Monica McGoldrick. (1990). « Génomogramme et entretien familial ». Paris : ESF.
- Gomez, P. A. et V. Richard Benjamins. (1999) « Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods » Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5) Stockholm, Sweden, August 2
- Gruber, Thomas R. (1993). « A translation approach to portable ontology specifications ». Journal of Knowledge Acquisition, 1993, vol. 5, 199-220.
- Gruber, Thomas R.(1995). «Toward Principles for the Design of Ontologies Used for Knowledge Sharing ». Journal of Human Computer Studies, vol. 43, 907-928.
- Grundstein, Michel.(2003) . « De la capitalisation des connaissances au management des connaissances dans l'entreprise, les fondamentaux du knowledge management ». Institut National des Télécommunications France. Entreprises : 3 jours pour faire le point sur le Knowledge Management. Avril 2003.
- Guarino, Nicola et Pierdaniele Giarretta (1995). « Ontologies and Knowledge Bases: Towards a Terminological Clarification ». Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing, Amsterdam: IOS Press. (Mars) P: 25-32.

- Guarino, Nicola. (1997). « Semantic Matching: Formal Ontological Distinctions for Information Organization, Extraction, and Integration. In Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology », SCIE 1997. Disponible sur le Web à l'adresse : <http://www.loa-cnr.it/Papers/KBKS95.pdf>
- Heijst Van G., A. T. Schreiber et B. J. Wielinga (1997). « Using explicit ontologies in KBS development ». International Journal of Human-Computer Studies, 45(2/3), 183–292.
- Holsapple, W. Clyde et K.D. Joshi. (2002) . «A Collaborative Approach to Ontology Design ». Communication of ACM, vol. 45, No. 2 (février).
- Jacob, Réal et Lucile Pariat. (2000). « Gérer les connaissances : un défi de la nouvelle compétitivité du 21^e siècle ». Disponible sur le Web à l'adresse : <http://www.cefric.gc.ca/rapports/gererconnaissance.pdf>.
- Kifer, Michael, Georg Lausen et James Wu. (1995). «Logical Foundations of Object-Oriented and Frame-Based Languages». Journal of the ACM (JACM), Vol. 42, Issue 4 (juillet) P: 741 - 843
- Lambert, Yves. (2006). Entrevues avec Monsieur Yves Lambert, Médecin de famille au Centre de Jeunesse de la Montérégie (Janvier-juin)
- Lefebvre Bernard , Serge Tadié, Omar Cherkaoui, Gilles Gauthier, Olivier Gerbé et Jean-Guy Meunier. (2003). « Le projet GDST », article Consulté le 15 janvier 2006 sur le Web : <http://www.gdst.ugam.ca/Documents/Articles/ArticleCometic2003.pdf#search=%22le%20projet%20GDST%22>.
- Maisondieu, Jean et Léon Métayer. (1986). « Les thérapies familiales ». Que sais-je? Collection encyclopédie, Paris.
- Mendes, Olavo, Valéry Psyché et Jacqueline Bourdeau. (2003). « État de l'art sur les méthodologies d'ingénierie ontologique ». Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, Vol.10, Montréal, QC.
- Murdock, George Peter. (1985). "Kin Term Patterns and their Distribution". World Cultures 1(4): stds25.dat, stds25.cod. Consulté le en avril 2006 sur le World Wide Web : <http://www.socialinfo.ch/cgi-bin/dicoposso/show.cfm?id=336>
- Natalya, F. Noy. (2004). « Semantic Integration: A survey Of Ontology-Based Approach». SIGMOD Record, Vol. 33, No. 4 (December).

- Natalya, F. Noy et Deborah L. McGuinness. (2003). « A Guide to Creating Your First Ontology ». Consulté le 15 avril 2006 sur le World Wide Web : <http://protege.stanford.edu/doc/users.html#tutorials>.
- Neches, Robert, Richard Fikes, Tim Finin, Thomas Gruber, Ramesh Patil, Ted Senator et William R. Swartout (1991). « Enabling Technology For Knowledge Sharing ». *AI Magazine*, Vol. 12, No. 3 (automne), p. 36-56.
- Nenad, Stojanovic, Jorge Gonzalez et Ljiljana Stojanovic. (2003). « ONTOLOGER – A System for Usage-Driven Management of Ontology-Based Information Portals » K-CAP'03. (Octobre) Sanibel Island, Florida, USA. Copyright 2003 ACM.
- Nguyen, Tho Hau et Yves Lambert. (1991). « GÉNI : le génogramme informatisé comme outil pour les professionnels de la santé ». Proposition de recherche UQAM,
- Nonaka, I. (1994). « Dynamic Theory of Organizational Knowledge Creation ». *Organizational Science*, Vol. 5, n°1 (février).
- Obrst, Leo. (2003). « Ontologies for Semantically Interoperable Systems » CIKM'03, November 3–8, 2003, New Orleans, Louisiana, USA. Copyright, ACM 1.
- OQLF : Office Québécois de la Langue Française. (2005). Disponible sur le Web à l'adresse : http://www.granddictionnaire.com/btml/fra/r_motclef/index1024_1.asp.
- Ouni, Ali et Aurélie Dudezert. (2004). « État de l'art des approches de définition du Système de Gestion des Connaissances ». Disponible sur le Word Wid Web à l'adresse : <http://www.knowledgeboard.com/cgi-bin/item.cgi?id=2253>.
- Prax, Jean-Yves. (2003). « Le manuel du knowledge management ». Paris, Dunod, 477 p.
- Rossett, A. (1999). « Knowledge management meets analysis ». *Training and Development*, vol. 53 (5), mai, p. 63-68.
- Rubenstein-Montano, B, Fay Leibowitz, Fudah Buchwalter, Doug McCaw, Butler Newman et Ken Rebeck. (2001) « SMARTVision: a Knowledge-management methodology » *Journal of Knowledge Management*. Vol. 5, No. 4, p. 300-310.
- Sogbohossou, Defodji (2002) « Ontologies, vocabulaires XML, représentation et exploitation des connaissances dans le cadre d'une organisation ». Mémoire de maîtrise en informatique de gestion, UQAM
- Sowa, John F. (2000). « Ontology, Metadata, and Semiotics ». *Proceedings of the International Conference on Conceptual Structures, ICCS*. Darmstadt, Germany. 14-18 Août. Disponible sur le Web à l'adresse : <http://www.bestweb.net/~sowa/peirce/ontometa.htm>.

- Studer, R., V.R. Benjamins et D. Fensel. « Knowledge Engineering, Principles and Methods » *Data and Knowledge Eng.*, Vol. 25, Mar. 1998, pp. 161-197.
- Swartout, Bill, Patil R., Knight K. et Russ T. (1997). « Towards Distributed Use of Large-Scale Ontologies » *Spring Symposium Series on Ontological Engineering*. p.138-148.
- Tourtier, P.-A. (1995). « Analyse préliminaire des métiers et de leurs interactions ». Rapport intermédiaire du projet GENIE, INRIA-Dassault-Aviation.
- Uschold, Mike, Martin King, Stuart Moralee et Yannis Zorgios. (1998). « The Enterprise Ontology ». *The Knowledge Engineering Review* No. 13: 31-89.
Disponible sur: <http://www.aiai.ed.ac.uk/project/enterprise/ontology.html>
- Uschold, Mike et Robert Jasper. (1999). « A Framework for Understanding and Classifying Ontology Applications » *Proceedings of the IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5) Stockholm, Sweden*, 2 aout
- Uschold, Michael et Michael Gruninger. (2004). « Ontologies and Semantics for Seamless Connectivity ». *ACM SIGMOD Record*, Vol. 33, No. 4 (Décembre).
- Wache, H., T. Vögel, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann et S. Hübner. (2001) . « Ontology-Based Integration of Information, A Survey of Existing Approaches ». *Proceeding of IGCAI-01 Workshop: Ontologies and information Sharing, Seattle, WA*, Vol.pp.108-117.
- W3C, (2004). <http://www.w3.org/2004/OWL/>
- Zacklad, Manuel et Michel Grundstein. (2001). « Management des connaissances » Paris : Hermès, 245 p.